

**DIEGO LAUCSEN DA ROSA**

**SISTEMA DE PROCESSAMENTO DE SINAIS  
BIOMÉDICOS: FILTRAGEM DE SINAIS DE  
ELETROENCEFALOGRAMA**

**FLORIANÓPOLIS - SC  
2009**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA**

**PROGRAMA DE PÓS-GRADUAÇÃO  
EM ENGENHARIA ELÉTRICA**

**SISTEMA DE PROCESSAMENTO DE SINAIS  
BIOMÉDICOS: FILTRAGEM DE SINAIS DE  
ELETROENCEFALOGRAMA**

Dissertação submetida à  
Universidade Federal de Santa Catarina  
como parte dos requisitos para a  
obtenção do grau de Mestre em Engenharia Elétrica.

**DIEGO LAUCSEN DA ROSA**

Florianópolis, Março de 2009.

# **SISTEMA DE PROCESSAMENTO DE SINAIS BIOMÉDICOS: FILTRAGEM DE SINAIS DE ELETROENCEFALOGRAMA**

Diego Laucsen da Rosa

'Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Engenharia Elétrica, Área de Concentração em Engenharia Biomédica, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.'

---

Fernando Mendes de Azevedo, D. SC.  
Orientador

---

Kátia Campos de Almeida, Ph.D.  
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

## **Banca Examinadora:**

---

Fernando Mendes de Azevedo, D. SC.  
Presidente

---

Fernanda Isabel Marques Argoud, Dr. Eng.

---

José Marino Neto, Dr.

---

André Luís Dalcastagnê, Dr. Eng.

---

Maria Nazaré Munari Angeloni Hahne, Dr.

## **DEDICATÓRIA**

Dedico às futuras gerações que utilizarão este trabalho.

## **AGRADECIMENTOS**

A meus pais, meus irmãos, meus amigos e a minha namorada, que muito me auxiliou no decorrer desta obra.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para obtenção do grau de Mestre em Engenharia Elétrica.

## **SISTEMA DE PROCESSAMENTO DE SINAIS BIOMÉDICOS: FILTRAGEM DE SINAIS DE ELETROENCEFALOGRAMA**

**Diego Laucsen da Rosa**

Março/2009

Orientador: Fernando Mendes de Azevedo, D. SC.

Co-Orientadora: Fernanda Isabel Argoud, Dr. Eng.

Área de Concentração: Engenharia Biomédica

Palavras-Chave: *Software* Livre, Plataforma Didática para Processamento de Sinais Biomédicos, Filtros Digitais, Processamento Digital de Sinais.

Número de Páginas: 135

A engenharia biomédica é uma área em fase de crescimento no Brasil. Em contrapartida, os cursos de formação em engenharia não formam um número suficiente de profissionais com os conhecimentos na área de engenharia biomédica. Para os poucos profissionais que aprendem assuntos sobre engenharia biomédica em sua formação, apenas uma pequena parte destas aulas foram de conteúdo prático. Assim, o presente trabalho detalha um estudo e desenvolvimento de uma ferramenta didática para apoio ao ensino de filtros digitais aplicados a sinais biomédicos. Processamento digital de sinais tem destaque como uma área complexa e que requer um alto conhecimento de ferramentas matemáticas para o seu projeto. Portanto, este trabalho propõe a criação de um tutorial interativo no qual os alunos possam ler sobre os assuntos relativos à filtragem digital e todo o conteúdo que introduz às técnicas de filtragem digital. O tutorial possui Java Applets e rotinas em Scilab que facilitarão o entendimento do conteúdo de maneira interativa, onde o leitor poderá ler e automaticamente executar o aprendido, tanto na teoria como na prática. Assim, com um tutorial teórico e prático, os alunos de Engenharia poderão, com mais facilidade, compreender assuntos relativos à filtragem digital e seus precedentes. O trabalho foi validado com a utilização de ferramentas matemáticas e avaliado com a utilização do WebMac (4.0), que com a alta expectativa de sucesso, os alunos poderão ter uma visão de como filtros digitais podem alterar um sinal de Eletroencefalograma, possibilitando esse aprendizado de forma estimulante e motivante.

Abstract of Dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

## **Biomedical Signal Processing System: Electroencephalogram Filtering System.**

**Diego Laucsen da Rosa**

March/2009

Advisor: Fernando Mendes de Azevedo, D. SC.

Co-Advisor: Fernanda Isabel Argoud, Dr. Eng.

Concentration Area: Biomedical Engineering.

Keywords: Free Software, Didactic Platform for Biomedical Signal Processing, Digital Filters, Digital Signal Processing.

Number of Pages: 135

The biomedical engineering is a growing area in Brazil though the Brazilian engineering courses do not graduate enough number of professionals with expertise in biomedical engineering field. For the few who raised these issues in their training, only a small portion of the classes were of practical content. Therefore, this present work details a study and development of a didactic tool to support the teaching of digital filters applied to biomedical signals. Digital signal processing is highlighted as a complex area and requires a high knowledge of mathematical tools for its project. This paper proposes the creation of an interactive tutorial, where students can read about the issues relating to digital filtering and all the content that introduces the techniques of digital filtering. The tutorial has Java Applets and routines in Scilab that facilitates the understanding of the content in an interactive way, where the reader is able to read and automatically run what was recently learned, both in theory and in practice. Thus, with a theoretical and practical tutorial, the students of engineering are able to easily understand issues on digital filtering and its precedents. The developed work was validated with the use of mathematical tools and evaluated with Webmac (4.0), that with the high expectation of success, the students can have a vision of how digital filters can distort an Electroencephalogram, allowing them to learn it in a stimulating and motivating way.

## SUMÁRIO

1 - INTRODUÇÃO.....	1
1.1 - O Eletroencefalograma.....	4
1.1.1 - Problemas gerais causados por Filtros.....	5
1.2 - Problema da Inserção da Engenharia Biomédica no Brasil.....	5
1.3 - Portal Saúde+EducaçãoTM.....	6
1.4 - Engenharia Biomédica e os Sinais Digitais.....	6
1.5 - Objetivo Geral.....	7
1.5.1 - Objetivos Específicos.....	7
1.6 - Justificativa.....	7
1.7 - Descrição do Conteúdo deste Trabalho.....	8
2 - PROCESSAMENTO DIGITAL DE SINAIS.....	9
2.1 - Apresentação.....	9
2.1.1 - Aplicações.....	9
2.2 - Sinais e Sistemas Discretos.....	10
2.2.1 - Sinais Discretos.....	10
2.2.2 - Definições e Propriedades de Sinais Discretos.....	12
2.2.3 - Sistemas Discretos.....	14
2.2.4 - Teorema de Nyquist.....	14
2.3 - Transformada Z.....	15
2.3.1 - A Transformada Z.....	15
2.3.2 - A Transformada Z Inversa.....	16
2.3.3 - Propriedades da Transformada Z.....	16
2.3.4 - Resposta em frequência de sistemas.....	19
2.3.5 - Resolução da Resposta em Frequência com aproximação Geométrica.....	20
2.3.6 - Equação de Diferenças.....	21
2.3.6.1 - Exemplo de como calcular os coeficientes da Equação de Diferenças.....	22
2.4 - Convolução Discreta.....	25
2.4.1 - Propriedades da Convolução.....	25
2.5 - Filtros Digitais.....	26
2.5.1 - Introdução a Filtros Digitais.....	26
2.5.2 - Histórico de Filtros Digitais.....	27
3 - FILTROS DE RESPOSTA FINITA AO IMPULSO (FIR).....	29
3.1 - Introdução.....	29
3.2 - Projeto de Filtros FIR.....	29
3.2.1 - Especificação do filtro.....	30
3.2.2 - Calcular os Coeficientes.....	31
4 - FILTROS DE RESPOSTA INFINITA AO IMPULSO (IIR).....	35
4.1 - Introdução.....	35
4.2 - Projeto de Filtros IIR com inserção de Polos e Zeros.....	36
4.2.1 - Exemplo da Utilização do método de inserção de polos e zeros para criação de um filtro Notch.....	36
4.3 - Projetos de Filtros IIR com a Transformação Bilinear Z (BZT).....	40
4.4 - Utilizando a Equação de Diferenças.....	42



5 - MATERIAIS E MÉTODOS.....	45
5.1 - Sistema de Processamento de Sinais Biomédicos.....	45
5.2 - Softwares Livres.....	45
5.2.1 - JUDE/Community e UML.....	45
5.2.2 - Desenvolvimento em JAVA.....	46
5.2.3 - Java Applets.....	46
5.2.3.1 - Componentes JAVA.....	47
5.2.4 - Scilab.....	47
5.3 - Validação das Ferramentas dos Tutoriais.....	48
5.3.1 - Transformada de Fourier.....	48
5.4 - Avaliação dos Tutoriais.....	48
5.4.1 - Webmac Senior (4.0)©.....	48
5.4.2 - Pontuação do WebMac Senior.....	49
5.4.3 - Classificação e Avaliação do Produto.....	50
5.4.4 - Questionário Aplicado.....	51
6 - DESENVOLVIMENTO.....	52
6.1 - Projeto.....	52
6.1.1 - Conteúdo Abordado.....	52
6.1.2 - Estruturação do Trabalho.....	53
6.1.3 - Modelagem.....	54
6.1.3.1 - Projeto de Filtros IIR com método de inserção de polos e zeros.....	56
6.1.3.2 - Projeto de Filtros FIR com métodos Janela.....	57
6.1.3.3 - Projeto de Filtros IIR com método da Transformação Bilinear.....	59
6.1.3.4 - Filtragem de Sinais utilizando coeficientes de filtros.....	61
6.1.4 - Implementação.....	62
6.1.4.1 - Implementação das Funções para o Scilab.....	62
6.1.4.2 - Implementação de Ferramentas de apoio para Aulas utilizando Scilab.....	65
6.1.4.3 - Implementação dos Java Applets.....	65
6.2 - Escrita dos Conteúdos.....	77
6.3 - Validação.....	77
6.3.1 - Validação das ferramentas para filtros FIR.....	78
6.3.1.1 - Passa Baixa.....	78
6.3.1.2 - Passa Alta.....	80
6.3.1.3 - Passa Banda.....	82
6.3.1.4 - Rejeita Banda.....	84
6.3.2 - Validação das ferramentas para filtros IIR.....	86
6.3.2.1 - Inserção de polos e zeros.....	86
6.3.2.2 - Transformação bilinear com Butterworth.....	89
6.3.2.3 - Transformação bilinear Z com Chebyshev.....	91
6.4 - Avaliação.....	93
6.4.1 - Alunos do Mini-Curso de Engenharia Biomédica na Prática.....	93
6.4.2 - Alunos de Sistemas Eletrônicos do IF-SC.....	95
6.5 - Resultados.....	96
7 - CONCLUSÃO.....	98
7.1 - Conclusão.....	98

7.2 - Trabalhos Futuros.....	99
ANEXO I – QUESTIONÁRIO WEBMAC 4.0 SENIOR.....	100
APÊNDICE I – DIAGRAMAS DE MODELAGEM.....	102
APÊNDICE II – TEXTO DE AJUDA DAS FUNÇÕES DO SCILAB.....	117
REFERÊNCIAS BIBLIOGRÁFICAS.....	134

## LISTA DE FIGURAS

Figura 1 - Sinal e EEG capturado de um Ser humano.....	2
Figura 2 - Sinal de EEG da figura 1 processado por um filtro Passa Baixas Freqüências.....	3
Figura 3 - Sinal de EEG da figura 2 filtrado por um filtro especial para remover a frequência da rede elétrica (60 Hz).....	4
Figura 4 - Sinal de Tempo Contínuo.....	11
Figura 5 - Sinal de Tempo Discreto.....	12
Figura 6 - Impulso Unitário.....	12
Figura 7 - Degrau Discreto.....	13
Figura 8 - Função Exponencial Real Discreta.....	14
Figura 9 - Círculo unitário no plano Z.....	16
Figura 10 - Relação entre o Plano S e o Plano Z.....	18
Figura 11 - Relação entre o Plano S e o Plano Z com hachuras.....	19
Figura 12 - Plano Z com um par de polos complexos conjugados e um par de zeros complexos e conjugados.....	23
Figura 13 - Máscara de um filtro passa baixa.....	30
Figura 14 - (a) Exemplo de um Passa Baixa ideal. (b) Gráfico dos coeficientes do filtro (Sinc).....	32
Figura 15 - Fórmulas para calcular os coeficientes de filtros.....	33
Figura 16 - Formulas para calcular alguns tipos de janelas.....	34
Figura 17 - Gráfico do plano Z com zeros em diferentes posições.....	37
Figura 18 - Gráfico do plano Z com polos em diferentes posições.....	38
Figura 19 - Gráfico do plano Z contendo um zero com coordenadas polar igual a 1 em 36°, e seu conjugado.....	39
Figura 20 - Gráfico do plano Z contendo um zero com coordenadas polar igual a 1 em 36 graus, e seu conjugado junto de um polo com coordenadas polares igual a 0,937 em 36 graus e seu conjugado.....	40
Figura 21 - Mapeamento do Plano S no Plano Z.....	41
Figura 22 - Gráfico do Efeito Warp.....	42
Figura 23 - Exemplo do Gráfico E.S.O.F.....	50
Figura 24 - Gráfico VXS.....	51
Figura 25 - Visão Geral do sistema.....	55
Figura 26 - Diagrama de Caso de Uso do PlanoZ.....	56
Figura 27 - Diagrama de Caso de Uso do FiltroFIR.....	58
Figura 28 - Diagrama de Caso de Uso do Caso " Calcular Coeficientes".....	59
Figura 29 - Diagrama de Caso de Uso do BZT.....	60
Figura 30 - Diagrama de Caso de Uso do FILTRO.....	61
Figura 31 - Tela Principal do Applet para filtros FIR.....	66
Figura 32 - Tela de configuração dos Filtros FIR.....	67
Figura 33 - Tela de Coeficientes do filtro FIR.....	68
Figura 34 - Tela Principal do PlanoZ.....	69
Figura 35 - Tela de configuração do PlanoZ.....	70
Figura 36 - Tela dos coeficientes do PlanoZ.....	71
Figura 37 - Tela principal da BZT.....	72

Figura 38 - Tela de configuração da BZT, antes da conversão.....	72
Figura 39 - Tela de configurações do BZT, depois da conversão.....	73
Figura 40 - Tela da Equação de Diferenças do BZT.....	74
Figura 41 - Tela principal com filtro criado e sinal carregado e filtrado.....	74
Figura 42 - Tela de criação de filtro.....	75
Figura 43 - Tela de carregamento de sinal no Applet.....	76
Figura 44 - Adição de frequências senoidais ao sinal original.....	77
Figura 45 - Resposta em Frequência do filtro passa baixa gerado com a função de gerar coeficientes para filtros FIR. Parte superior a resposta proporcional, na parte inferior a resposta em Decibéis.....	79
Figura 46 - Sinal do EEG sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.....	79
Figura 47 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.....	80
Figura 48 - Resposta em Frequência do filtro passa alta gerado com a função de gerar coeficientes para filtros FIR. Parte superior a resposta proporcional, na parte inferior a resposta em Decibéis.....	81
Figura 49 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo, o sinal filtrado.....	81
Figura 50 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e abaixo, a FFT após a aplicação do filtro.....	82
Figura 51 - Resposta em Frequência do filtro passa Banda gerado com a função de gerar coeficientes para filtros FIR. Na parte superior, a resposta proporcional, na parte inferior, a resposta em Decibéis.....	83
Figura 52 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.....	83
Figura 53 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.....	84
Figura 54 - Resposta em Frequência do filtro rejeita banda gerado com a função de gerar coeficientes para filtros FIR. Na parte superior, a resposta proporcional, na parte inferior, a resposta em Decibéis.....	85
Figura 55 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.....	85
Figura 56 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e abaixo a FFT após a aplicação do filtro.....	86
Figura 57 - Polos e Zeros de um filtro Butterworth Passa Alta, no plano Z.....	87
Figura 58 - Configurações do filtro em questão.....	87
Figura 59 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.....	88
Figura 60 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.....	88
Figura 61 - Polos e Zeros de um filtro Butterworth Passa Baixa, no plano S.....	89
Figura 62 - Configurações do filtro em questão.....	90
Figura 63 - Sinal sendo filtrado pelo filtro em questão. A cima o sinal natural e abaixo o sinal filtrado.....	90

Figura 64 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. A cima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.....	91
Figura 65 - Polos e Zeros de um filtro Chebyshev Tipo I Passa Baixa, no plano S.....	92
Figura 66 - Configurações do filtro em questão.....	92
Figura 67 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. A cima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.....	93
Figura 68 - Gráfico E.S.O.F. do questionário aplicado à turma do segundo Mini-curso de Engenharia Biomédica na Prática. E para Estimulante, S para Significativo, O para Organizado e F para Fácil-de-Usar.....	94
Figura 69 - Gráfico VXS da turma que respondeu o formulário no Segundo Mini-Curso de Engenharia Biomédica na Prática.....	94
Figura 70 - Gráfico E.S.O.F. do questionário aplicado à turma do IF-SC. E para Estimulante, S para Significativo, O para Organizado e F para Fácil-de-Usar.....	95
Figura 71 - Gráfico VXS da turma que respondeu o formulário do IF-SC.....	96

## 1 - INTRODUÇÃO

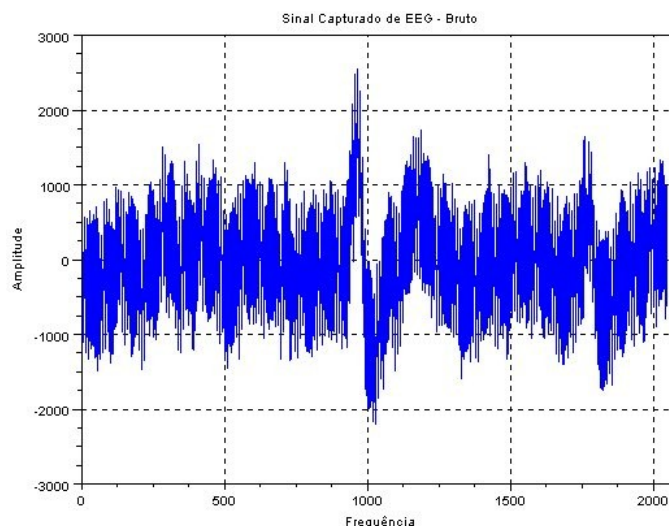
Desde todo conhecimento obtido pela humanidade em relação à eletricidade e após as realizações de experiências tentando medir os potenciais elétricos do corpo humano, foi em meados de 1875 e 1877 que Richard Caton realizou os primeiros experimentos que mediu os potenciais elétricos do cérebro de animais, como ratos, coelhos e macacos. (Diedermeyer E. e Da Silva F. L.,2004, pg.2).

Caton utilizou um Galvanômetro para seus experimentos. Inserindo os sensores, um na maçã cinzenta do cérebro e o outro no crânio, ele notou que a tensão variava com o passar do tempo. Com ajuda de uma agulha, que vibrava conforme a tensão do Galvanômetro e pintava o caminho percorrido em uma folha que se deslocava constantemente, Canton pode observar o Eletroencefalograma destes animais. (Diedermeyer E. e Da Silva F. L.,2004, pg. 2)

Eletroencefalograma (EEG), etimologicamente, é formado por três palavras: **eletro** (do grego *élektron*) exprime a idéia de Eletricidade; **encéfalo** (do grego *egképhalos*) representa o cérebro e grama (do grego *grámma*) exprime a idéia de letra, escrita, figura, desenho, etc. Assim, eletroencefalograma representa idéia de um registo das atividades elétricas do cérebro, assim como os experimentos que Cantos efetuou.

Na época de Canton, o sinal era capturado e o resultado desenhado em um folha contínua. Para estudar estes sinais, era necessário analisar estes resultados desenhados nas folhas. Hoje em dia, com a discretização dos sinais, é possível amostrar estes sinais e estudá-los em um computador, podendo usufruir de todas as ferramentas que a informática nos fornece, desde computadores pessoais, até micro-controladores especiais para trabalhar com sinais.

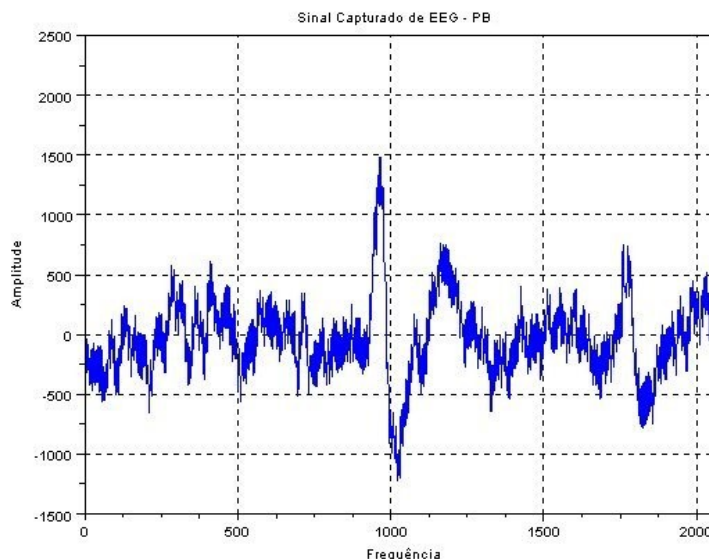
Seguindo os passos de Canton, porém utilizando eletrodos de EEG, um amostrador digital e um computador para visualizar um sinal, poderíamos chegar a algo como a Figura 1.



*Figura 1 - Sinal e EEG capturado de um Ser humano.*

Na Figura 1, podemos notar um sinal ruidoso e sem (ou com poucas) informações úteis para um especialista. Este tipo de ruído, do tipo branco, é geralmente causado por componentes de alta frequência e quando afeta um sinal, e sua amplitude for maior que a amplitude das frequências existentes no sinal, ele pode o distorcer completamente.

Então, por se tratar de um ruído de alta frequência, seria possível com a existência de um filtro eliminar as altas frequências e ficar apenas com as baixas. Se efetuássemos este processo no sinal da Figura 1, obteríamos algo como o sinal representado na Figura 2.

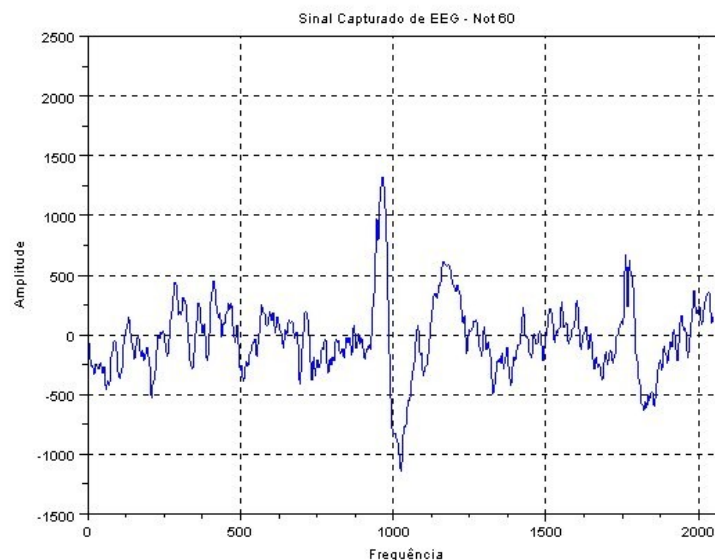


*Figura 2 - Sinal de EEG da figura 1 processado por um filtro Passa Baixas Frequências.*

Observando a Figura 2, o sinal sem as componentes de alta frequência se tornou mais claro, porém ainda possui algum ruído ou interferência.

O aparelho de captura do sinal, no momento da captura, poderia estar em um lugar passível de interferência, por exemplo, a frequência presente na rede elétrica. Então, com um filtro que filtre estas frequência, poderia-se encontrar um sinal mais limpo, conforme a Figura 3.





*Figura 3 - Sinal de EEG da figura 2 filtrado por um filtro especial para remover a frequência da rede elétrica (60 Hz).*

Na Figura 3, é possível observar um sinal de Eletroencefalograma de um ser humano pronto para análise. É desta maneira que um especialista em analisar estes sinais o deseja e o utilizará para realizar um exame.

## 1.1 - O Eletroencefalograma

A fisiologia explica detalhadamente o funcionamento do EEG, em resumo, ele é gerado pelos impulsos elétricos causados pelos potenciais de ação dentro dos neurônios. A soma destes impulsos (positivos e negativos, conforme o sentido da corrente) podem ser medidos e essa tensão colhida e registrada consiste o EEG.

Os sinais de EEG são utilizados para diversos fins, como por exemplo, detecção de padrões cerebrais que podem levar a patologias, como Epilepsia e doenças do Sono.

Como todo sinal, o EEG é formado por uma soma de frequências. Quando aplica-se um filtro a um EEG, deseja-se remover ou modificar certas componentes de frequências deste sinal, como por exemplo, remover um artefato, linha de Base ou alguma interferência (Conforme demonstrado na seção anterior). Porém, quando executa-se uma operação destas, para manter a qualidade e a integridade do sinal, é necessária cautela em não remover frequências importantes do sinal. Frequências que se forem filtradas ou distorcidas podem levar o especialista a um diagnóstico

errado sobre um fato específico que o mesmo esteja estudando.

### 1.1.1 - Problemas gerais causados por Filtros

O sinal de EEG é composto basicamente por quatro padrões rítmicos: as ondas delta, theta, alpha e beta. As ondas delta se caracterizam por constituírem da soma de frequências abaixo de 4 Hz. As ondas theta são constituídas por frequências entre 4 Hz e 8 Hz. As ondas alpha possuem frequências entre 8 e 13 Hz e as ondas beta possuem ondas menores que 13 Hz.

Sendo assim, filtros projetados sempre precisam levar em consideração que para filtrar um sinal mantendo a fidelidade da informação, o filtro não deve modificar as frequências faixa entre 0 Hz e 13 Hz.

Em geral, há duas maneiras de modificar estas frequências: uma delas é modificar a amplitude do sinal e a outra a fase. E são estes cuidados que os projetistas precisam ter na hora de se projetar um filtro para EEG.

## 1.2 - Problema da Inserção da Engenharia Biomédica no Brasil

Recentes estudos sobre a inclusão da Engenharia Biomédica (EB) no meio acadêmico tem mostrado uma grande ausência de disciplinas e conhecimento de Engenharia Biomédica apresentado nos cursos de formação da área, como cursos de Engenharia Elétrica, Engenharia Eletrônica etc.

De acordo com os estudos realizados pelo Instituto de Engenharia Biomédica, da Universidade Federal de Santa Catarina, no artigo publicado na Revista Brasileira de Engenharia Biomédica, “A inserção da Engenharia Biomédica nos cursos de graduação de Engenharia Elétrica no Brasil” (POSSA, P. R. et al., 2008); dos 326 cursos de Engenharia Elétrica, de acordo com os critérios descritos no artigo, apenas 214 disponibilizam informações sobre sua grade curricular. Destes 214 cursos, apenas 16 possuem disciplinas de Engenharia Biomédica em sua grade curricular. Destes 16 cursos, no total foram identificadas 40 disciplinas, que variam entre as diversas áreas da Engenharia Biomédica. Destas 40 disciplinas, apenas 5 são obrigatórias e das 40 disciplinas, que formam um conteúdo de 2.208 horas aula, apenas 14,0% (310 horas-aula) possuem uma abordagem prática. Nos 86,0% (1.898 horas-aula) restantes, os temas de EB são abordados de forma exclusivamente teórica.

Este fato mostra que a Engenharia Biomédica está fracamente presente no meio acadêmico em forma teórica e muito menos de maneira prática. Porém, a Engenharia Biomédica é uma área

em ascensão no Brasil e novas ferramentas de apoio ao ensino poderiam contribuir na formação desta área. Assim, o Instituto de Engenharia Biomédica-UFSC investiu no “Sistema de Processamento de Sinais Biomédicos (SPSB)”, que constitui em um conjunto de ferramentas para ensino de diversos assuntos que giram em torno da EB, com abordagem teórica e prática.

O SPSB é suportado pelo Portal Saúde+Educação<sup>TM</sup>, e inserido no SPSB, está o “Filtragem de sinais de Eletroencefalograma”, que é formado pelo conteúdo deste trabalho.

### 1.3 - Portal Saúde+Educação<sup>TM</sup>

O Portal Saúde+Educação<sup>TM</sup> ([www.saudeeducacao.ufsc.br](http://www.saudeeducacao.ufsc.br)) é um sistema desenvolvido pelo IEB-UFSC para suporte à educação na área da saúde (incluindo Engenharia Biomédica), que possibilita a administração de tutoriais e outros sistemas.

O sistema de gerenciamento de Tutoriais permite que os documentos sejam importados em HTML com suporte a imagens em diversas formas. Também é possível anexar arquivos para *download* nas páginas.

O sistema já suporta, desde que configurado anteriormente, a possibilidade de separar os conteúdos em páginas, para que futuramente, quando utilizado, os usuários possam navegar entre os conteúdos de forma prática e de fácil uso.

### 1.4 - Engenharia Biomédica e os Sinais Digitais

Incluído nos assuntos abordados pela Engenharia Biomédica, tem destaque a filtragem digital de sinais. A filtragem digital não é só um assunto presente na Engenharia Biomédica como também nas Engenharias em geral. Para o projeto e utilização de filtros digitais, há a necessidade de um vasto conhecimento matemático e o uso de ferramentas complexas.

Este trabalho propõe o desenvolvimento de um módulo de Filtragem Digital (Filtragem de sinais de Eletroencefalograma), inserido no SPSB. Este módulo consiste em um conjunto de tutoriais para o ensino de projetos e utilização de filtros digitais de sinais, aplicados especificamente a sinais de eletroencefalograma. Os tutoriais contam com o apoio de JAVA Applets<sup>1</sup> e rotinas em Scilab<sup>2</sup> que facilitam o entendimento do conteúdo de uma maneira interativa,

---

1 JAVA Applets: Aplicativo que é compilado e executado em tempo de execução junto de um navegador Web. Ele permite que um pequeno software seja aberto na janela do navegador e possa executar funcionalidades como um *software* normal.

2 Scilab: *Software* livre para realização de cálculos matemáticos. Versão similar ao MatLab, porém em versão gratuita.

onde os leitores podem aprender inicialmente na teoria e consecutivamente visualizar o que aprendeu na prática. E, finalmente, o leitor poderá aprender como deve proceder para manter a integridade de um sinal de Eletroencefalograma, possibilitando-os a aprender de uma maneira estimulante e motivante.

## 1.5 - Objetivo Geral

O objetivo geral do presente trabalho é a implementação do tutorial “Filtragem de Sinais de Eletroencefalograma”, a ser disponibilizado no Portal Saúde+Educação<sup>TM</sup>, usando JAVA Applets, rotinas em Scilab e ferramentas de *software* livre, para sinais bioelétricos, focando na apresentação de características de fase e amplitude da transformada de Fourier de sinais de eletroencefalograma, e tendo como público alvo alunos de graduação em Engenharia Elétrica com conhecimento de sinais e sistemas lineares e análise em frequência e sinais analógicos contínuos.

### 1.5.1 - Objetivos Específicos

- Implementação de um conjunto de ferramentas em Scilab para projetos de filtros digitais. Estas ferramentas incluem a possibilidade de visualização de polos e zeros no plano Z, magnitude e fase da resposta do sinal. Ela também possibilitará que o usuário teste o filtro com um sinal que deseje e observe os resultados e por fim possa extrair os coeficientes para a construção do filtro. Por fim, devem possuir a possibilidade de desenvolvimento de alguns modelos analógicos na forma digital, como Butterworth, Chebyshev tipo 1 e tipo 2 e Elíptico em diversas ordens e demonstrar os problemas de fase e amplitude relacionados a filtragem digital de Eletroencefalograma.
- Desenvolver Applets para apoio ao ensino teórico sobre Filtragem Digital de Sinais.
- Avaliação e validação com a aplicação de questionários para a turma do 2º Minicurso de Engenharia Biomédica e uma turma do curso Superior de Tecnologia de Sistemas Eletrônicos do IF-SC.

## 1.6 - Justificativa

A área de Processamento Digital de Sinais tem se destacado como uma área complexa e que requer um alto conhecimento de ferramentas matemáticas para o seu projeto, tais como:

- Compreensão dos sinais no domínio do tempo e da frequência;

- Conhecimento e aplicação de transformadas Z e convoluções;
- Entender os resultados de uma transformada de Fourier;
- Compreender a formação do plano Z.

Por isto, este projeto sugeriu a criação de um tutorial interativo, em que o usuário pode ler sobre os assuntos relativos à filtragem digital e todo o conteúdo que introduz as técnicas de filtragem digital. Estes tutoriais possuem JAVA Applets e rotinas em Scilab que facilitam o entendimento do conteúdo de maneira interativa, onde o leitor pode ler e automaticamente executar o aprendizado, tanto na teoria como na prática.

Estes tutoriais foram inicialmente preparados em Scilab e em seguida em Java Script, para que fossem suportados pelo padrão da Plataforma de Ensino em Engenharia Biomédica, que se constitui de tutoriais em HTML.

Assim, com um tutorial teórico e prático, os alunos de Engenharia Elétrica podem com mais facilidade compreender os assuntos relativos à Filtragem Digital e seus precedentes. Enfim, podem ter uma visão de como estes filtros devem ser projetados para manter a qualidade e o valor informativo do sinal de Eletroencefalograma devido a amplificação e a defasagem de frequências.

## **1.7 - Descrição do Conteúdo deste Trabalho**

O capítulo 2 apresenta a fundamentação teórica sobre Processamento Digital de Sinais e suas ferramentas mais utilizadas. Esta teoria é utilizada como base para filtragem de sinais digitais.

O capítulo 3 traz com mais detalhes as teorias sobre filtros de Resposta Finita ao Impulso e descreve com detalhes o método de Janelas.

O capítulo 4 traz com mais detalhes as teorias sobre filtros de Resposta Infinita ao Impulso e descreve com detalhes dois métodos para sua utilização.

O capítulo 5 descreve os materiais e métodos utilizados no desenvolvimento deste trabalho.

O capítulo 6 descreve o desenvolvimento do projeto e implementação.

Por fim, o capítulo 7 descreve as conclusões deste trabalho.

## 2 - PROCESSAMENTO DIGITAL DE SINAIS

Um sinal de eletroencefalograma digitalizado é poluído por ruídos e interferência. Este sinal pode ser processado por um sistema que opera como um filtro para o ruído, possibilitando a visualização do sinal sem interferências e ruídos. Este capítulo aborda o básico sobre processamento digital de sinais e as ferramentas necessárias para o seu projeto e aplicação.

### 2.1 - Apresentação

A área de processamento digital de Sinais está presente nas tecnologias que se expandem em diversas áreas, como entretenimento, comunicação, exploração espacial, medicina, arqueologia etc. (A. V. Openheim, 1999, p.31.).

Processamento de sinais é considerada a transformação ou manipulação das informações que um sinal contém. Por exemplo, separar dois sinais que anteriormente foram unidos. (A. V. Openheim, 1999, p.31.). A diferença entre o processamento de sinais contínuos e discreto, é que o sinal discreto está digitalizada ou seja, composto por um conjunto de valores, consecutivos, que foram anteriormente amostrados ou gerados por um dispositivo digital.

Neste capítulo, serão descritos dados fundamentais dos sinais discretos e como funcionam, até ir de encontro com os filtros digitais, que são um dos tipos de processamento digital de sinais.

#### 2.1.1 - Aplicações

- Telecomunicações: a telefonia tradicional (analógica) foi sendo substituída rapidamente pela atual (digital) e ambas ligada à rede de telecomunicação digital. Assim, computadores ligados em pontos diferentes dessa grande rede são usados para o processamento de informações e dados. (B. A. Sheno, 2006, p. 1-2).
- Processamento de Voz: processamento Digital de Sinais é usado para a melhoria da qualidade da voz em sistemas digitais de telefonia fixa ou móvel. Também é usado para processamento de reconhecimento de voz, tradução de texto para voz (*text-to-speech*) ou de voz para texto (*speech-to-text*). (B. A. Sheno, 2006, p. 2).
- Aparelhos Eletrônicos: diversos aparelhos eletrônicos utilizam o processamento digital de sinais, como por exemplo os celulares, computadores, aparelhagem de edição de som, a nova linha de Televisão Digital etc. (B. A. Sheno, 2006, p. 2). Nos microcontroladores destes aparelhos eletrônicos, são utilizadas funções de Processamento Digital de Sinais

como: análise de espectro, redução de ruído e compressão de dados etc. (E. C. Ifeachor; B. W. Jervis, 1993, p. 3).

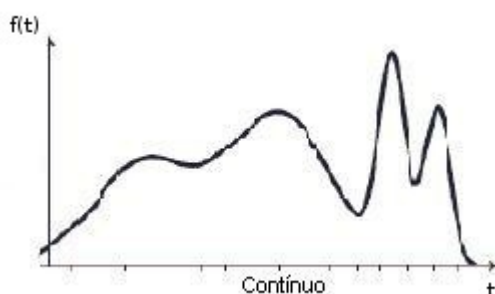
- Sistemas Biomédicos: diversos aparelhos presentes no ambiente hospitalar utilizam o processamento digital de sinais, como monitores de Eletroencefalograma e Eletrocardiograma e também para o processamento de imagens médicas, como Ultrassom e Ressonância Magnética etc. (B. A. Shenoi, 2006, p. 2).
- Processamento de Imagens: exemplo de processamento de imagens são rotinas usadas para melhoramento de imagens, restauração de imagens, entendimento de certas imagens, radar e sonar (B. A. Shenoi, 2006, p. 2), reconhecimento de padrões, visão robótica, reconhecimento de imagens, previsão de clima com visão via satélites e animações. (E. C. Ifeachor; B. W. Jervis, 1993, p. 3).
- Eletrônica Militar: processamento digital de sinais é bem usado na área militar, como por exemplo: rastreamento com GPS, radar e sonar, guia de projéteis, detecção e rastreamento de alvos, comunicações seguras e criptografia, controle remoto etc. (B. A. Shenoi, 2006, p. 2-3).
- Aeronáutica e Aeroespacial: processamento digital de sinais é usado para monitoramento e controle da performance do voo, controle de funcionamento e segurança de motores de voo, comunicação, análise de vibração etc. (B. A. Shenoi, 2006, p. 3).
- Aplicação Industrial: robótica, produção em massa, segurança, controle numérico, vídeo conferência etc. (B. A. Shenoi, 2006, p. 3).

## **2.2 - Sinais e Sistemas Discretos**

### **2.2.1 - Sinais Discretos**

Um sinal define a variação de uma grandeza física como função de uma ou mais variáveis e esta variação contém a informação que é importante (B. A. Shenoi, 2006, p. 4). A idéia de se processar um sinal é usar o sinal original e modificá-lo de alguma maneira. Por exemplo, para somar dois sinais sonoros em um único sinal, como em uma música com dois instrumentos, é necessária uma relação direta entre eles, que no caso são as variáveis dependentes e independentes que em ambos sinais são iguais, neste caso, o tempo e a amplitude. Assim, somariam-se as amplitudes, mas o tempo seria exatamente o mesmo para os dois sinais.

Um sinal “contínuo no tempo” é um sinal que contém no mínimo uma variável independente que é contínua. Um sinal contínuo no tempo de uma dimensão pode ser representado por uma função no tempo que varia continuamente de  $-\infty$  até  $+\infty$ . Mas pode ser em função de outras variáveis, como temperatura, pressão etc., que no caso não são necessariamente contínuas. Então o sinal é contínuo no tempo quando a variável independente é contínua, mas em diferentes valores da variável contínua a variável dependente pode ser descontínua. (B. A. Sheno, 2006, p. 4). Uma função no tempo é representada tradicionalmente por  $f(t)$ . Um exemplo de um sinal contínuo no tempo pode ser visto na Figura 4.



*Figura 4 - Sinal de Tempo Contínuo.*

Um sinal “discreto” é uma função que é definida apenas em instantes discretos do tempo e indefinido em qualquer outro instante de tempo. (B. A. Sheno, 2006, p. 4). Assim, em uma medida de uma variável dependente do tempo no intervalo entre  $-\infty$  até  $+\infty$ , serão considerados apenas alguns valores da variável dependente, não todos (uma vez que o tempo é discreto). Então, enquanto no sinal “contínuo no tempo” temos o tempo  $t$ , no sinal “discreto” temos  $nT$ , onde  $T$  é um intervalo de tempo conhecido e constante em segundos que se chama **taxa de amostragem** e  $n$  um valor inteiro que pode variar de  $-\infty$  até  $+\infty$ . Assim, para cada intervalo de tempo dado por  $T$ , captura-se um valor da variável dependente e numeramos essa amostra com  $n$ . No fim da captura, tem-se um conjunto cheio de valores que representam o sinal capturado. Um sinal discreto é representado tradicionalmente por  $f(n)$ . Um exemplo de um sinal discreto pode ser visualizado na Figura 5.



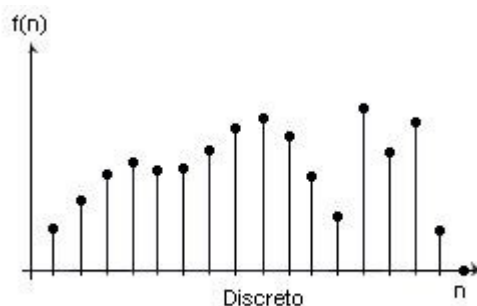


Figura 5 - Sinal de Tempo Discreto.

### 2.2.2 - Definições e Propriedades de Sinais Discretos

Função Impulso Unitário: A função impulso  $\delta(n)$  é definida por:

$$\delta[n] = \begin{cases} 0, & n \neq 0 \\ 1, & n = 0 \end{cases} \quad 1$$

A importância da expressão 1, do impulso unitário, para os sistemas discretos da mesma maneira que o impulso unitário no tempo contínuo é importante para os sistemas de tempo contínuo, isto é: possuindo um sistema linear e invariante no tempo e aplicando este sistema e impulso unitário a operação convolução, resulta em um sinal idêntico ao sistema. Isto ocorre porque o impulso causa resposta em todas as frequências, identificando o comportamento do sistema a todas as frequências possíveis.

A Figura 6 demonstra um exemplo de função impulso unitário discreto.

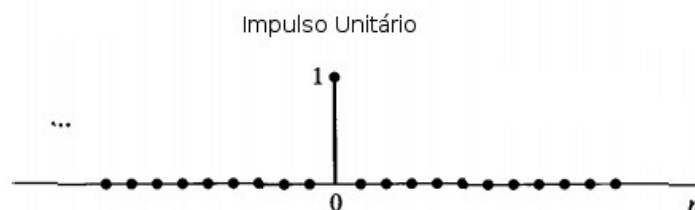


Figura 6 - Impulso Unitário.

Fonte: Adaptado de OPENHEIM (1999).

Função Degrau: A função degrau  $u(n)$  é definida por:

$$\delta[n] = \begin{cases} 0, & n < 0 \\ 1, & n \geq 0 \end{cases} \quad 2$$

A expressão 2 constrói um degrau conforme Figura 7.



*Figura 7 - Degrau Discreto.*

*Fonte: Adaptado de OPENHEIM (1999).*

Uma das importâncias da função degrau é que ela pode representar o momento em que um sistema está ligado ou desligado, e também os momentos em que o sistema está ligando e desligando.

Por exemplo, a entrada de um sistema encontra-se em zero (momento em que a função degrau ainda é zero). No momento em que o sistema é ligado, a função de entrada passa para valor um e permanece em um até ser desligada. Quando desligada o valor da entrada passa para zero.

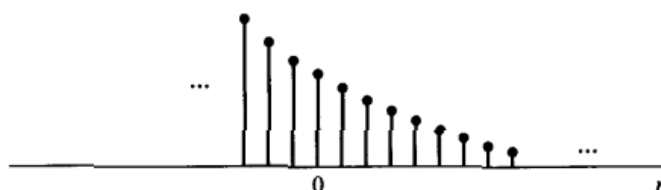
Um exemplo prático: O lugar está frio, logo alguma pessoa aperta o botão ligar do aquecedor. Neste instante, a entrada passou de zero para um e o ambiente começa a esquentar, lentamente. Quando a pessoa nota que o ambiente ficou quente demais, ela desliga o aquecedor. Esta entrada entre zero e um pode ser representada pela função degrau.

Função Exponencial Real: Uma função Exponencial Real Discreta é representada pela expressão 3:

$$x[n] = A\alpha^n \quad 3$$

onde  $A$  é uma constante real. Se  $A$  for complexa, então se torna uma função exponencial complexa.

A Figura 8 demonstra o resultado da expressão 3 com um valor de  $A$  qualquer.



*Figura 8 - Função Exponencial Real Discreta.*

*Fonte: Adaptado de OPENHEIM (1999).*

A função exponencial tem um valor muito importante na representação e na análise de sistemas lineares e invariantes no tempo.

### 2.2.3 - Sistemas Discretos

Sistemas discretos são processos que atuam sobre os sinais discretos, recebendo um ou mais sinais discretos de entrada e fornecendo como saída um ou mais sinais discretos. Em geral, todos os sistemas implementados por um computador ou microcontrolador são sistemas discretos. (Cadzow e Martens, 1970, p 4).

### 2.2.4 - Teorema de Nyquist

O teorema de Nyquist diz que: “a condição para um sinal íntegro é que a maior frequência no sinal de entrada de um amostrador analógico/digital de um sistema deve ser igual ou menor que a metade da taxa de amostragem”.

Se esta regra for obedecida na amostragem, o sinal é amostrado fielmente e sem distorções. Caso contrário, se na entrada do sistema fossem observadas frequências maiores que a metade da taxa de amostragem, o sinal será deformado de uma maneira irreversível. Esse tipo de problema é chamado de *aliasing*, que vem a alterar o sinal, que jamais poderá ser recuperado com fidelidade, em todos os casos se tornando inútil.

Para isto, existe o conceito de filtro anti-*aliasing*, que é um filtro analógico inserido

defronte ao conversor de maneira a filtrar as frequências maiores que a metade da taxa de amostragem, para que seja captada apenas a região de interesse e assim obter um sinal fiel.

## 2.3 - Transformada Z

A transformada Z é uma ferramenta útil para representação, análise e projeto de sistemas discretos. Ela tem a mesma função no sistema discreto que a transformada de Laplace tem para o sistema contínuo no tempo. (E. C. Ifeachor; B. W. Jervis, 1993, p. 103).

Este tópico introduz a transformada Z e suas aplicações.

### 2.3.1 - A Transformada Z

A transformada Z é definida pela expressão 4:

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \quad 4$$

onde  $x(n)$  é uma sequência numérica a ser transformada e  $z^{-n}$  uma variável complexa.

Em sistemas causais, a equação da transformada Z é reduzida para:

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n} \quad 5$$

A transformada Z tem semelhanças com a transformada de Laplace, porém em domínios diferentes. Também, a transformada Z está mais próxima da transformada de Fourier, pela semelhança da expressão.

Uma vez que Z é uma variável complexa, a transformada Z precisa ser representada no plano complexo. No plano Z, o mapeamento de  $Z=1$  se dá em um círculo de raio 1, conforme a Figura 9.

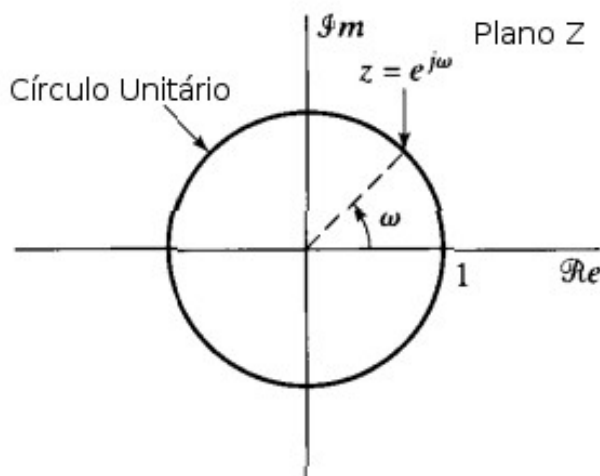


Figura 9 - Círculo unitário no plano Z.

Fonte: Adaptado de OPENHEIM (1999).

No Plano Z, conforme mostra Figura 9, que também pode ser chamado de Plano Complexo Z, é onde os sistemas discretos são representados, por polo e zeros, de maneira semelhante da transformada de Laplace.

### 2.3.2 - A Transformada Z Inversa

A transformada Z inversa permite encontrar a sequência discreta original dada uma sequência transformada previamente. (E. C. Ifeachor; B. W. Jervis, 1993, p. 109). A sua fórmula é a seguinte:

$$x(n) = Z^{(-1)}[X(z)] \quad 6$$

onde  $X(z)$  é a transformada Z de  $x(n)$  e  $Z^{-1}$  é o símbolo da transformada Z inversa.

### 2.3.3 - Propriedades da Transformada Z

- Linearidade: Se existem duas sequências discretas,  $x_1(n)$  e  $x_2(n)$ , e ambas possuem respectivamente suas Transformadas Z,  $X_1(z)$  e  $X_2(z)$ , então a combinação linear se dá por:

$$a x_1(n) + b x_2(n) \rightarrow a X_1(z) + b X_2(z) \quad 7$$

- Deslocamento: Se existe a transformada Z de uma sequência  $x(n)$  que é  $X(z)$  e a transformada Z desta sequência for atrasa em  $m$  amostras, então este atraso é de  $X^{-m}(z)$ .

$$x(n) \rightarrow X(z)$$

$$x(n-m) \rightarrow z^{-(n-m)} X(z) \quad 8$$

- Convolução: Dado um sistema linear e invariante no tempo,  $x(n)$ , e uma resposta um pulso,  $h(n)$ , definido por:

$$y(n) = \sum_{k=-\infty}^{\infty} h(k) x(n-k) \quad 9$$

Possui a transformada Z:

$$Y(z) = H(z) X(z) \quad 10$$

onde  $X(z)$ ,  $H(z)$  e  $Y(z)$  são, respectivamente, as transformadas Z de  $x(n)$ ,  $h(k)$  e  $y(n)$ .

Isto ocorre porque a convolução no domínio do tempo corresponde a uma multiplicação no domínio Z.  $H(z)$  também pode ser reconhecida como a função de transferência do domínio Z.

- Diferenciação: Se  $X(z)$  é a transformada Z de  $x(n)$ , a transformada Z de  $nx(n)$  pode ser obtida diferenciando  $X(z)$ :

$$x(n) \rightarrow X(z)$$

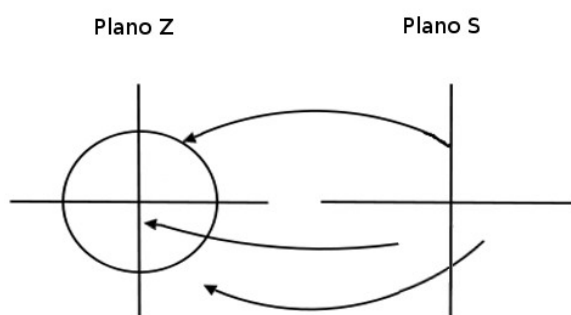
$$nx(n) \rightarrow -z \frac{dX(z)}{dz} \quad 11$$

onde  $dx$  é a derivada da função.

Esta propriedade é útil quando deseja-se descobrir a transformada inversa de sistemas com polos de múltiplas ordens. (E. C. Ifeachor; B. W. Jervis, 1993, p. 128)

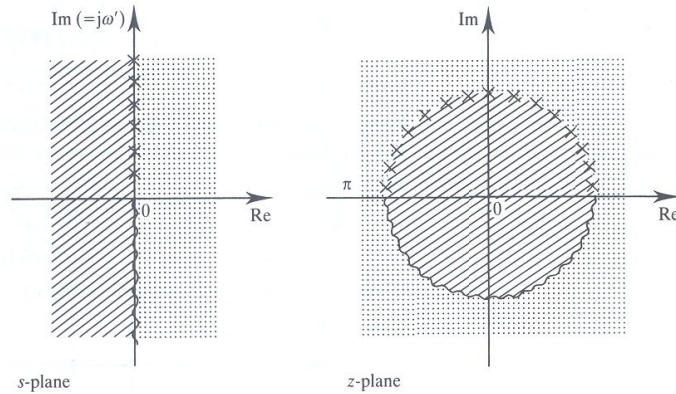
- Relação com a Transformada de Laplace: A transformada Z representa para os sistemas discretos exatamente o que a Transformada de Laplace representa para os sistemas de

tempo contínuo, porém, a transformada de Laplace transfere os dados para o plano S e a Transformada Z para o plano Z. Considerando a diferença entre um sistema de tempo contínuo e um de tempo discreto, Ifeachor e Jervis (1993, p. 129) mostra na seguinte figura a relação que os domínios têm entre si.



*Figura 10 - Relação entre o Plano S e o Plano Z.*

Visualizando a Figura 10, é possível visualizar superficialmente a relação entre os dois planos. Tudo que no plano S está mapeado à esquerda do eixo central encontra-se no plano Z dentro do círculo unitário. À direita do eixo central do plano S é mapeado fora do círculo unitário e o que está sobre o eixo imaginário no plano S é mapeado sobre o círculo unitário. Na Figura 11 pode-se visualizar a mesma transformação com hachuras. A Figura 11 mostra este mapeamento com hachuras.



*Figura 11 - Relação entre o Plano S e o Plano Z com hachuras.*

### 2.3.4 - Resposta em frequência de sistemas

Para descobrir qual a resposta para cada frequência em um sistema discreto, utiliza-se a seguinte fórmula:

$$H(Z) = \sum_{n=-\infty}^{\infty} h(n) z^{-n} \quad 12$$

e aplica-se a seguinte substituição:

$$z = e^{(j\omega T)} \quad 13$$

e finalmente simplificando as duas equações:

$$H(e^{(j\omega T)}) = \sum_{n=-\infty}^{\infty} h(n) e^{(-jn\omega T)} \quad 14$$

Com esta substituição, pode-se variar o ângulo  $\omega$  de 0 até  $\pi$  e pode-se plotar em um gráfico para visualizar qual é a resposta para cada frequência do sistema. Combinando este gráfico e o Teorema de Nyquist, sabemos que o ponto inicial de  $\omega$  (0 rad) equivale a 0 Hz e para o valor  $\pi$  rad teremos o equivalente a metade da Taxa de amostragem.

O próximo tópico defronta com uma seguinte pergunta: “Como criar uma rotina computacional para a visualização da resposta em frequência?”. Criar uma rotina com a equação acima é consideravelmente complicado, para facilitar existe o Método Geométrico para Aproximação da Transformada Z, que é apresentado a seguir.



### 2.3.5 - Resolução da Resposta em Frequência com aproximação Geométrica

Este método é uma maneira fácil e simples de descobrir a resposta em frequência de um sistema discreto, porém, para executá-lo é necessário conhecer todos os polos e zeros do sistema.

Com a função de transferência do sistema, pode-se extrair as raízes do polinômio do numerador e do denominador. As raízes do numerador são os zeros do sistema, as do denominador são os polos do sistema. A equação a seguir exemplifica melhor essa visualização. (E. C. Ifeachor; B. W. Jervis, 1993, p. 135).

$$H(z) = \frac{K(z-z_1)(z-z_2)\dots(z-z_N)}{(z-p_1)(z-p_2)\dots(z-p_M)} = \frac{\prod_{i=1}^N K(z-z_i)}{\prod_{k=1}^M (z-p_k)} \quad 15$$

Onde K é o ganho proporcional do sistema,  $z_1, z_2 \dots z_N$  são os zeros do sistema,  $p_1, p_2, \dots p_M$  são os polos do sistema e z é a variável independente do sistema.

Observando a expressão 15, fica mais fácil entender que cada parêntese no conjunto representa uma entidade (polo/zero), conforme sua localização (denominador/numerador).

A partir da expressão 15, simplifica-se e chega-se a expressão geral, que é apresentada pela expressão 16:

$$H(e^{j\omega T}) = \frac{\prod_{i=1}^N K(e^{j\omega T} - z_i)}{\prod_{k=1}^M (e^{j\omega T} - p_k)} \quad 16$$

Agora, assumindo um sistema com 2 polos e 2 zeros para exemplificar, faz-se uma substituição na expressão 16:

$$H(e^{j\omega T}) = \frac{K(e^{j\omega T} - z_1)(e^{j\omega T} - z_2)}{(e^{j\omega T} - p_1)(e^{j\omega T} - p_2)} \quad 17$$

Como sabe-se que  $e^{j\omega T}$  é um ângulo, é necessário uma simplificação que dará o resultado com fórmula final para este cálculo:

$$H(e^{j\omega T}) = \frac{K U_1 \theta_1 U_2 \theta_2}{V_1 \phi_1 V_2 \phi_2} \quad 18$$

Onde U é distância entre o zero e o ponto onde  $\omega T$  se encontra, V a distância entre o polo e o ponto

onde  $\omega T$  se encontra,  $\theta$  é o ângulo formado entre o U e a referência X e  $\Phi$  é o ângulo formado entre o V e a referência X.

Primeiro, aplicando-se  $e^{j\omega T}$  na equação, depois o módulo e em seguida multiplica-se pelo número imaginário correspondente que representa a entidade do plano (polo/zero). Com estes passos, pode-se notar que as componentes se simplificam, então para o cálculo da resposta em frequência para magnitude, o que interessa é apenas a distância (V ou U), resultando em uma fórmula mais simplificada:

$$H(e^{j\omega T}) = \frac{U_1 U_2}{V_1 V_2} \quad 19$$

considerando  $K = 1$ .

E no cálculo da resposta em frequência para fase, as componentes se simplificam e resta apenas um somatório do ângulo dos zeros subtraído do somatório dos ângulos da fase, conforme:

$$H(e^{j\omega T}) = \theta_1 + \theta_2 - (\phi_1 + \phi_2) \quad 20$$

Assim, tem-se duas expressões, uma para a magnitude e outra para a fase com operações geométricas, que torna simples a implementação. Este cálculo, implementado em *software*, torna-se um cálculo iterativo, onde o tempo de processamento irá definir a resolução e a qualidade da resposta.

### 2.3.6 - Equação de Diferenças

A equação de diferenças é um conjunto de valores que representa um sistema discreto no domínio do tempo contínuo. Aplicando-se uma convolução entre a equação de diferenças e um sinal de entrada no domínio no tempo contínuo, resulta em um sinal de saída também no tempo contínuo (E. C. Ifeachor; B. W. Jervis, 1993, p. 143). Isso quer dizer, tendo-se um sistema discreto descrito por um vetor de números que representam um sistema no domínio discreto (Z) e um outro vetor que representa um sinal amostrado no domínio do tempo (t) e com este sistema e sinal é possível calcular a saída do sistema. Isto é feito com um processo que permite calcular a equação de diferença através da função de transferência do sistema no plano Z.

Considerando o caso simples, onde a equação da função de transferência é definida apenas por um polinômio de  $n$  ordem sem a utilização de outros componentes (como seno e cosseno, logaritmo e exponencial). Neste caso, a equação de transferência pode ser definida conforme:

$$y(n) = \sum_{k=0}^N a_k x(n-k) - \sum_{k=1}^M b_k y(n-k) \quad 21$$

onde  $x(n)$  é a amostra de entrada,  $y(n)$  a de saída,  $y(n-k)$  são as saídas passadas e  $a_k$  e  $b_k$  são os coeficientes que representam o sistema. A equação 21 demonstra que a saída atual é obtida através da entrada atual e passadas e saídas passadas.

Entretanto, a equação 21 demonstra como certos coeficientes são utilizados para aplicar o sistema discreto em um sinal amostrado no domínio do tempo. Agora, é necessário descobrir como calcular estes coeficientes ( $a_k$  e  $b_k$ ).

Inicia-se observando como as partes da equação 21 pode ser relacionada a Transformada Z.

$$a_k x(n) = a_k X(z)$$

$$a_k x(n-k) = a_k z^{(-k)} X(z)$$

Estas são duas comparações feitas com dois blocos que juntos podem representar a expressão 21. Pode-se notar que as igualdades alcançam outros blocos já vistos anteriormente a na formulação da Transformada Z. Logo, poderemos regredir a equação de diferenças para a função de transferência do domínio Z e associando os coeficientes a alguma propriedades do domínio.

Então, a equação 21, com as substituições, pode ser escrita conforme:

$$Y(z) = \sum_{k=0}^N a_k z^{-k} X(z) - \sum_{k=1}^M b_k z^{-k} Y(z) \quad 22$$

E simplificando a equação 22, teremos a equação de transferência do sistema discreto:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^N a_k z^{(-k)}}{1 + \sum_{k=0}^M b_k z^{(-k)}} \quad 23$$

Finalmente, para encontrar os coeficientes ( $a_k$  e  $b_k$ ), é necessário apenas igualar a função de transferência conforme a igualdade da expressão 23. Para utilizar este processo, foi descrito o exemplo a seguir.

### 2.3.6.1 - Exemplo de como calcular os coeficientes da Equação de Diferenças

Usando o exemplo de um sistema contendo dois polos e dois zeros, sendo eles:

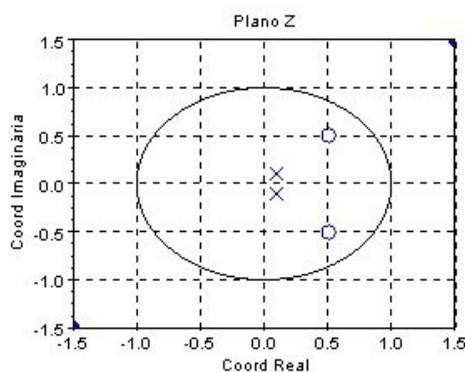
$$\text{Zero } 1 = z_1 = 0,5 + 0,5j$$

$$\text{Zero } 2 = z_2 = 0,5 - 0,5j$$

$$\text{Pólo } 1 = p_1 = 0,1 + 0,1j$$

$$\text{Pólo } 2 = p_2 = 0,1 - 0,1j$$

As imagens dos polos e zeros no domínio  $Z$  são semelhantes ao domínio  $S$ , onde são descritos por polos e zeros sendo números imaginários. Também, os polos e zeros sempre são apenas descritos na parte superior do círculo unitário e sempre possuem um conjugado, que decorre devido ao teorema de Nyquist. Haverá situações em que a entidade estará sobre o eixo  $x$ , então, por não possuir a parte imaginária, não haverá a necessidade de demonstrar o seu conjugado. Os polos e zeros acima se posicionam de acordo a Figura 12:



*Figura 12 - Plano Z com um par de polos complexos conjugados e um par de zeros complexos e conjugados.*

*Resultado apresentado pelo Software Scilab.*

Conforme equação 23, os zeros encontram-se na parte superior da divisão e os polos na inferior, expandindo o polinômio dos zeros e dos polos separadamente, partindo da expressão 16.

$$H(z) = \frac{K(z-z_1)(z-z_2)\dots(z-z_N)}{(z-p_1)(z-p_2)\dots(z-p_N)}$$

$$\frac{K(z-z_1)(z-z_2)}{(z-p_1)(z-p_2)}$$

Fazendo as substituições e considerando  $K = 1$ .

$$H(z) = \frac{(z-0,5+0,5j)(z-0,5-0,5j)}{(z-0,5+0,1j)(z-0,1-0,1j)}$$

$$\frac{z^2 + 0,5z + 0,5jz - 0,5z + 0,25 + 0,25j - 0,5jz - 0,25j - 0,25j^2}{z^2 + 0,1z + 0,1jz - 0,1z + 0,01 + 0,01j - 0,1jz - 0,01j - 0,01j^2}$$

Com

$$j = \sqrt{-1}$$

Temos;

$$\frac{z^2 + 0,5z + 0,5 \quad jz - 0,5z + 0,25 + 0,25 \quad j - 0,5 \quad jz - 0,25 \quad j + 0,25}{z^2 + 0,1z + 0,1 \quad jz - 0,1z + 0,01 + 0,01 \quad j - 0,1 \quad jz - 0,01 \quad j + 0,01}$$

Agora simplificando;

$$\frac{z^2 + 0,5}{z^2 + 0,02} \quad 24$$

Na equação 24, pode-se observar que, de acordo com os princípios da transformada Z, o exponencial  $z^2$  está buscando valores futuros na série de entrada ou saída.

Quando um sistema depende de valores futuros ele é dito não causal, ou seja, o cálculo da saída se dá antes mesmo que a entrada ocorra no sistema. Como não é possível medir valores futuros em um sistema em tempo-real, efetua-se uma transformação matemática para converter todos os valores para que o sistema dependa apenas de valores passados, tornando-o causal.

$$\begin{aligned} & \frac{z^2 + 0,5}{z^2 + 0,02} \cdot \frac{z^{-2}}{z^{-2}} \\ & \frac{1z^0 + 0,5z^{-2}}{1z^0 + 0,02z^{-2}} \end{aligned} \quad 25$$

E assim associando 25 a 23:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1z^0 + 0,5z^{-2}}{1z^0 + 0,02z^{-2}}$$

$$Y(z) * 1z^0 + 0,02z^{-2} = X(z) * 1z^0 + 0,5z^{-2}$$

$$Y(z) 1z^0 + Y(z) 0,02z^{-2} = X(z) 1z^0 + X(z) 0,5z^{-2}$$

$$1y(n-0) + 0,02y(n-2) = 1x(n-0) + 0,5x(n-2)$$

$$y(n) + 0,02y(n-2) = x(n) + 0,5x(n-2)$$

E sabendo que  $y(n)$  é a saída do sistema, pose-se isolá-la para encontrar qual a saída presente do sistema em determinado momento:

$$y(n) = -0,02y(n-2) + x(n) + 0,5x(n-2)$$

Com esta expressão, finalmente ela pode ser aplicada em um sinal amostrado no tempo. Caso implementado em *software* para realizar esta tarefa automaticamente, o algoritmo precisaria de ao menos quatro valores de memória, para armazenar duas entradas passadas [ $y(n-1)$  e  $y(n-2)$ ] e

duas saídas passadas  $[x(n-1)$  e  $x(n-2)]$ . No caso,  $x(n-1)$  e  $y(n-1)$  não são utilizados na expressão, mas são armazenados para que no próximo instante eles se transformem em  $x(n-2)$  e  $y(n-2)$ , respectivamente.

Com este trecho pode-se ter uma idéia de como funciona o mecanismo de se aplicar um sistema discreto a um sinal no tempo.

## 2.4 - Convolução Discreta

O termo convolução descreve como uma entrada de um sistema interage com o sistema e produz uma saída. (E. C. Ifeachor; B. W. Jervis, 1993, p. 213). Geralmente, o sinal vai ser modificado de alguma maneira produzindo uma saída, que é uma versão alterada da entrada.

Então, com esse breve conceito, podemos imaginar que quando temos um sistema discreto e um sinal discreto, podemos executar a convolução entre os dois e obter uma saída, que seria exatamente a saída do sistema para aquele sinal de entrada.

A fórmula que descreve a convolução discreta é a seguinte:

$$y(n) = \sum_{m=0}^n h(n-m)x(m) \quad 26$$

sendo  $h(n-m)$  a série percorrendo o conjunto de dados que representa o sistema,  $x(m)$  o sinal de entrada e  $y(n)$  a entrada no tempo  $n$ . A convolução também pode ser representada com o operador  $(*)$  (asterisco entre parênteses ou circulado).

$$y = h(*)x \quad 27$$

### 2.4.1 - Propriedades da Convolução

- Comutativa: Sinal  $x_1$  convoluido a  $x_2$  deve possuir o mesmo resultado se  $x_2$  for convoluido a  $x_1$ .

$$x_1(*)x_2 = x_2(*)x_1 \quad 28$$

- Distributiva:

$$x_1(*)[x_2 + x_3] = [x_1(*)x_2] + [x_1(*)x_3] \quad 29$$

- Associativa:

$$x_1(*)[x_2(*)x_3] = [x_1(*)x_2](*)x_3 \quad 30$$

## 2.5 - Filtros Digitais

Este tópico introduz filtros digitais, com conhecimento necessário para as próximas seções que se aprofundarão em dois tipos de filtros digitais, os filtros FIR e os filtros IIR.

### 2.5.1 - Introdução a Filtros Digitais

Um filtro é essencialmente um sistema que seletivamente modifica a forma de onda de um sinal, sendo na amplitude e/ou na fase de uma ou mais frequências. (E. C. Ifeachor; B. W. Jervis, 1993, p. 252). Ou seja, um filtro tem a capacidade de remover (reduzindo amplitude a zero, ou quase) certas frequências de um sinal. Com essa afirmação, podemos notar que podemos criar um filtro para remover a frequência dos ruídos de um sinal ou de uma interferência específica.

Um filtro digital é uma rotina matemática implementada em *hardware* ou *software* que opera sobre um sinal discreto, produzindo uma saída discreta que será utilizado a um propósito. O termo digital surge do fato de o filtro ser implementado em *software* ou *hardware*, ou seja, em meios digitais.

Segundo Ifeachor e Jervis (1993, p. 252-253), um filtro digital apresenta as seguintes vantagens e desvantagens:

Principais vantagens:

- Filtros digitais tem certas características que não são possíveis em filtros analógicos.
- Diferentemente de filtros analógicos, filtros digitais não variam com mudanças ambientais, como variação de temperatura. Isto elimina a necessidade de calibragem constante.
- A resposta em frequência de um filtro digital pode ser ajustada automaticamente se ele for implementado utilizando um processador programável, que é o caso de um filtro adaptativo, que se auto corrige com o tempo.
- Muitos canais de entrada podem ser filtrados por um único filtro sem a necessidade de replicação do *hardware*.
- Os dados não filtrados e filtrados podem ser armazenados para uma utilização, caso desejado. (Considerando que quando isto é feito em um filtro analógico, o processo é mais complexo que para o digital).
- O limite de cálculo está na resolução da palavra do microcontrolador, que permite atenuações menores que 70 db. (Os autores descreveram isto em 1993, quando ainda não se



imaginava um DSP de 32 bits ou um computador de 64 bits.).

- Vantagens são obtidas quando implementadas em microcontroladores e DSPs (Digital Signal Processor), que têm uma gama de séries projetadas para a execução em baixo consumo e ocupando muito pouco espaço, o que possibilita a implementação destes filtros digitais internamente aos DSPs.
- Os microcontroladores que trabalham em alta frequência, possibilitando milhões de instruções por segundo, permitem a implementação de algoritmos e *softwares* complexos e com velocidade de processamento suficiente. (Por exemplo, em sistemas controlados com microcontroladores para prevenção de danos causados pelos terremotos).
- O tamanho dos microcontroladores nos dias de hoje permitem a construção de dispositivos móveis de tamanho compacto, valorizando os dispositivos móveis.

### 2.5.2 - Histórico de Filtros Digitais

Inicialmente, as teorias de sinais discretos foram motivadas para o cálculo de integral e interpolação e resolver equações diferenciais, porém, como os computadores ainda não existiam, a teoria foi considerada inútil até a chegada dos computadores. Então, a solução para os sistemas físicos modelados por equações diferenciais foram implementados da forma digital. (B. A. Shenoi, 2006, p. 20).

Com o tempo, as teorias sobre filtros digitais foram amadurecendo, e a criação do teorema Nyquist-Shannon e o algoritmo para transformada rápida de Fourier (por Cooley-Tukey) formaram os fundamentos do processamento digital de sinais.

Nos dias de hoje, filtros digitais têm uma grande importância. Em quase todos os lugares do mundo existe um aparelho elétrico e/ou eletrônico. E quando se trata de um aparelho para manipulação de sinais, este aparelho pode possuir internamente um filtro digital.

Na informação digital, uma vez que um sinal tenha sido amostrado, ele não está mais passível de interferência eletromagnética ou ruídos. Então, os filtros digitais surgem como uma alternativa que traz muitas vantagens em relação aos filtros analógicos.

Em contrapartida, os computadores tiveram que evoluir rapidamente para suportar certos processamentos, pois substituir um filtro de um sistema que trabalha com altas frequências requer um computador que possa manipular aquela informação na mesma velocidade que o filtro analógico.

Porém, com o surgimento de DSPs (Digital Signal Processor), que são computadores dedicados para processamento de sinais em alta frequência, os filtros digitais vem ganhando cada vez mais espaço no mercado mundial.

### 3 - FILTROS DE RESPOSTA FINITA AO IMPULSO (FIR)

Este capítulo descreve o que são os Filtros de Resposta Finita ao Impulso, suas características, como projetá-los e utilizá-los.

#### 3.1 - Introdução

Primeiro será feita uma breve descrição das características principais dos filtros FIR, de acordo com Ifeachor e Jervis (1993, p. 279):

- Um filtro FIR básico é caracterizado pelas duas equações seguintes:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad 31$$

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k} \quad 32$$

onde  $h(k)$  são os coeficientes de “resposta ao impulso” do sistema,  $H(z)$  é a função de transferência do sistema discreto (no domínio  $Z$ ),  $N$  é o tamanho do filtro e  $x$  é o sinal de entrada. A equação 31 é a equação de diferenças do filtro FIR. Ela está descrita no domínio do tempo e na forma não recursiva, de maneira a utilizar apenas valores presentes e passados de  $x(n)$ . A equação 32 é a função de transferência do filtro.

- Filtros FIR tem a fase da resposta em frequência linear. (Discutido a seguir).
- Filtros FIR são simples de serem implementados e os *Digital Signal Processor*<sup>3</sup> (DSP) tem ferramentas prontas para otimizar a utilização de filtros FIR.

#### 3.2 - Projeto de Filtros FIR

Para se projetar um filtro FIR é necessário executar os seguintes passos:

- Especificação do filtro: este tópico inicia com a escolha do tipo do filtro e suas características.
- Calcular os Coeficientes: os coeficientes do filtro (sistema) são calculados. Este sistema irá respeitar cada característica determinada na primeira etapa.

E a seguir uma descrição de cada tópico e como executar cada etapa.

---

3 *Digital Signal Processor*: Processador Digital de Sinais (PDS).

### 3.2.1 - Especificação do filtro

As especificações de um filtro se dão através de um modelo, onde desenhamos a característica fundamental do filtro e depois ajustamos valores que são requisitos para o filtro que está sendo projetado.

Por exemplo, um filtro passa baixa possui uma máscara conforme a Figura 13:

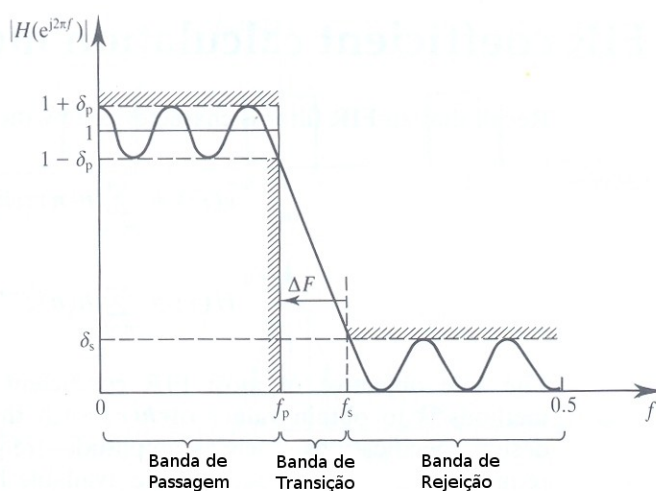


Figura 13 - Máscara de um filtro passa baixa.

E neste modelo, alguns requisitos são especificados conforme o desejo do projetista. Dentre eles, estão desenhados na Figura 13:

- $\delta_p$  : Amplitude dos *ripples* na banda de passagem.
- $\delta_s$  : Amplitude máxima dos *ripples* na banda de rejeição.
- $f_p$  : Frequência limite da banda de passagem.
- $f_s$  : Frequência limite da banda de rejeição.

Também é importante para o projeto, mas não está incluso na figura:

- $F_s$  : Frequência de Amostragem.

Estes são cinco atributos geralmente encontrados em projetos de filtros. Geralmente, estes valores em si são as especificações. Por exemplo um projetista poderia dizer:

“Filtro passa baixa com atenuação máxima de 1,3 dB na banda de passagem, frequência de

corte em 200 Hz, com início da banda de rejeição em 250 Hz e atenuação mínima de -46 dB na banda de rejeição e possuindo uma taxa de amostragem de 1 kHz”. Um filtro assim teria as seguintes características:

- $\delta_p : 0,07 = (1,3\text{dB} \approx 20 \cdot \log(1 + 0,07) = 20 \cdot \log(1 + \delta_p))$ .
- $\Delta_s : 0,1 = (20 \cdot \log(0,1) \approx -46\text{dB})$ .
- $f_p : 0,2$
- $f_s : 0,25$
- $F_s : 1000$

Conhecendo todos os atributos do filtro, pode-se partir para a etapa de calcular os coeficientes que caracterizem o filtro projetado.

### 3.2.2 - Calcular os Coeficientes

Relembrando as equações 31 e 32, o método de calcular os coeficientes consiste em de alguma maneira calcular o  $h(n)$ , ou pelo menos uma aproximação, para que este sistema se comporte como um filtro previamente projetado. Existem diversas maneiras para se conseguir isso, porém escolhida foi o método da Janela. O método da janela foi escolhido por ser o mais simples de se implementar e por ser o mais usado em projetos de filtros digitais.

O Método das Janelas consiste em desenhar um filtro ideal no domínio  $Z$  e em seguida executada a transformada  $Z$  inversa sobre este filtro projetado. Com isso tem-se os coeficientes prontos para serem utilizados como filtro, porém uma gama de problemas surge no caminho. A solução que dá o nome a este método é demonstrada a seguir.

Por exemplo, quando projeta-se um filtro passa baixa, imagina-se um passa-baixa ideal como na Figura 14.a:

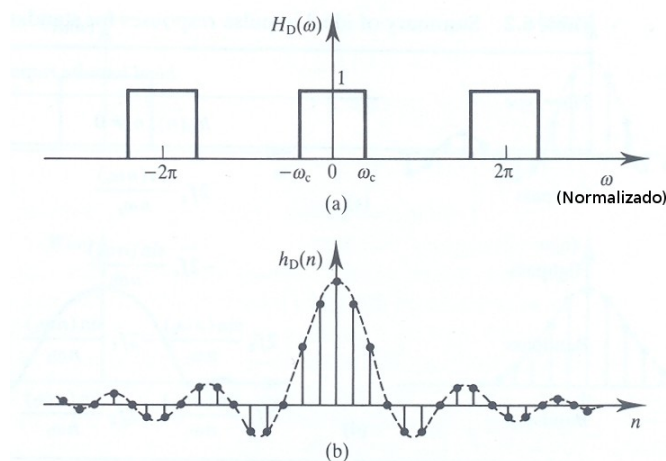


Figura 14 - (a) Exemplo de um Passa Baixa ideal. (b) Gráfico dos coeficientes do filtro (Sinc).

Dessa maneira seria óbvio: Passa 100% abaixo da frequência de corte e atenua 100% das frequências acima da frequência de corte. Então, para calcularmos os coeficientes deste filtro, precisa-se aplicar a transformada Z inversa, que resulta em um vetor de coeficientes representado pela Figura 14.b:

A Figura 14.b, que são os coeficientes de resposta ao impulso de um passa-baixa, podem ser calculados com a seguinte fórmula:

$$h_D(n) = \frac{2 f_c \sin(n \omega_c)}{n \omega_c}, n \neq 0, -\infty \leq n \leq \infty \quad 33$$

E com  $n = 0$  a função torna-se indeterminada, por teorema de L'Hôpital podemos calcular que, para  $n = 0$  a função fica:

$$h_D(n) = 2 f_c, n = 0, -\infty \leq n \leq \infty \quad 34$$

A Figura 15, mostra como projetar outros filtros:

Tipo do Filtro	Resposta ao Impulso Ideal, $h_D(n)$	
	$h_D(n), n \neq 0$	$h_D(0)$
Passa Baixa	$2f_c \frac{\sin(n\omega_c)}{n\omega_c}$	$2f_c$
Passa Alta	$-2f_c \frac{\sin(n\omega_c)}{n\omega_c}$	$1 - 2f_c$
Passa Banda	$2f_2 \frac{\sin(n\omega_2)}{n\omega_2} - 2f_1 \frac{\sin(n\omega_1)}{n\omega_1}$	$2(f_2 - f_1)$
Rejeita Banda	$2f_1 \frac{\sin(n\omega_1)}{n\omega_1} - 2f_2 \frac{\sin(n\omega_2)}{n\omega_2}$	$1 - 2(f_2 - f_1)$

$f_c, f_1$  e  $f_2$  são as frequências limite da banda de passagem e rejeição; N é o número de coeficientes do filtro.

Figura 15 - Fórmulas para calcular os coeficientes de filtros.

Ai surge um problema que ocorre, se  $n$  tende do  $-\infty$  até o  $+\infty$ , como pode-se escrever toda a função para filtrar um sinal? Em um sistema, seriam necessárias infinitas áreas de memória para armazenar todos os dados em memória.

A única solução é truncar o vetor em algum ponto em direção a  $-\infty$  e em algum ponto em direção a  $+\infty$ . Então, geralmente realizam-se a seguinte regra:  $h_D(n) = 0$  quando  $n$  for maior ou menos que  $M$ . Entretanto, isto causa diversos *ripples* e *overshots* indesejáveis. Isto é conhecido como fenômeno de Gibbs e para amenizar este problema, utiliza-se uma janela, que dá o nome ao método.

A janela ( $w(n)$ ) consiste em um vetor do mesmo tamanho do vetor de coeficientes ( $h_D(n)$ ) o qual cada posição será multiplicada pela posição correspondente do vetor de coeficientes:

$$H_D(n) = h_D(n) * w(n), -M \leq n \leq M \quad 35$$

Então, um filtro sem janela é na verdade uma janela em que todas as posições são iguais a 1(um), chamada de janela Retangular.

O que a Janela traz de melhoramento ao filtro, é que algumas janelas, projetadas adequadamente, fazem com que os *ripples* e os *overshots* sejam distribuídos nas outras regiões do filtros (banda de passagem e banda de rejeição).

A Figura 16 traz uma lista com as janelas mais utilizadas e seus efeitos sobre o filtro:

Nome da função Janela	Tamanho da Banda de Transição (Hz) (Normalizado)	Ripples na banda de Passagem (dB)	Atenuação na Banda de Rejeição (dB) (Máxima)	Função Janela $w(n),  n  \leq (N-1)/2$
Retangular	$0.9/N$	0.7416	21	1
Hanning	$3.1/N$	0.0546	44	$0.5 + 0.5 \cos\left(\frac{2\pi n}{N}\right)$
Hamming	$3.3/N$	0.0194	53	$0.54 + 0.46 \cos\left(\frac{2\pi n}{N}\right)$
Blackman	$5.5/N$	0.0017	74	$0.42 + 0.5 \cos\left(\frac{2\pi n}{N-1}\right) + 0.08 \cos\left(\frac{4\pi n}{N-1}\right)$
Kaiser	$2.93/N$ ( $\beta = 4.54$ )	0.0274	50	$\frac{I_0(\beta\{1 - [2n/(N-1)]^2\}^{1/2})}{I_0(\beta)}$
	$4.32/N$ ( $\beta = 6.76$ )	0.00275	70	
	$5.71/N$ ( $\beta = 8.96$ )	0.000275	90	

*Figura 16 - Formulas para calcular alguns tipos de janelas.*

Observando na Figura 16, podemos notar que as janelas causam outros efeitos que podem fazer com que o filtro desrespeite as características iniciais. Então estes valores devem ser considerados. Por exemplo, o campo “Tamanho da Área de Transição” vai definir o número de coeficientes de seu filtro. “Ripple na banda de passagem” tem que estar de acordo com o projetado e etc.

Então, com estes dados, tem-se quatro passos necessários para o cálculo dos coeficientes:

1. Especificar o tipo de resposta em frequência ideal do filtro projetado: escolher a fórmula que associa ao filtro que se está projetando (Passa Alta, Passa Baixa, Passa Banda ou Rejeita Banda).
2. Recolher os dados da máscara necessários para aplicar na fórmula da Figura 16.
3. Escolher a janela e o tamanho do vetor de coeficientes para que o filtro respeite as condições iniciais.
4. Com todos estes dados calcular  $w(n)$ ,  $h_D(n)$  e finalmente multiplicando os dois vetores se obtém  $H_D(n)$ , conforme equação 35.

Seguindo esses passos os coeficientes de um filtro podem ser facilmente calculados. Enfim, de acordo com o visto, para utilização deste filtro, apenas é necessário aplicar a convolução de  $H_D(n)$  com algum sinal e resposta em frequência do filtro se aplicarão no sinal, filtrando-o.



## 4 - FILTROS DE RESPOSTA INFINITA AO IMPULSO (IIR)

Este capítulo descreve o que são os Filtros de Resposta Infinita ao Impulso, suas características, como projetá-los e utiliza-los.

### 4.1 - Introdução

OS filtros IIR são caracterizados pelas seguintes equações:

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^N a_k x(n-k) - \sum_{k=1}^M b_k y(n-k) \quad 36$$

onde  $h(k)$  é a resposta ao impulso do filtro que é teoricamente infinita,  $a_k$  e  $b_k$  são os coeficientes do filtro e  $x(n)$  e  $y(n)$  são a entrada e saída do filtro. Resumindo, temos a partir da equação 36 e função de transferência do sistema:

$$H(z) = \frac{a_0 + a_1 z^{-1} + \dots + a_N z^{-N}}{1 + b_1 z^{-1} + \dots + b_M z^{-M}} = \frac{\sum_{k=0}^N a_k z^{-k}}{1 + \sum_{k=1}^M b_k z^{-k}} \quad 37$$

Uma parte importante dos projetos de filtros IIR é encontrar bons valores para os coeficientes  $a_k$  e  $b_k$  de maneira que as características do filtro, como resposta em frequência, se comportem de maneira correta. (Ifeachor e Jervis, 1993, p. 375).

Note que, na equação 36, uma parte da equação ( $y(n-k)$ ), depende de valores de saídas passadas, e também de entradas presentes e passadas (na parte  $x(n-k)$ ), que é o que caracteriza os filtros IIR. Uma das vantagens do filtro IIR é que ele precisa de menos coeficientes que o FIR para representar um bom filtro, reduzindo o processamento necessário, porém traz desvantagens, como a facilidade em perder a estabilidade.

A equação 37 pode ser distribuída da seguinte maneira:

$$H(z) = \frac{K(z-z_1)(z-z_2)\dots(z-z_N)}{(z-p_1)(z-p_2)\dots(z-p_M)} \quad 38$$

onde  $z_1, z_2, \dots, z_N$  são os zeros do sistema e  $p_1, p_2, \dots, p_M$  são os polos do sistema.

O diagrama onde os polos e os zeros são plotados no plano  $Z$  e representado pelo círculo unitário, que se torna uma boa ferramenta para análise de estabilidade de filtros. Algumas regras conhecidas

podem ser observadas, como por exemplo: para a estabilidade do filtro, um polo deve sempre estar dentro do círculo unitário e se sobre o círculo, precisa estar junto de um zero. Já os zeros não possuem restrições.

Para se projetar um filtro IIR, existem diversos métodos. Aqui serão abordados os métodos:

- Inserção de Polos e Zeros;
- Transformação Bilinear Z.

O Método de Inserção consiste em inserir os polos e zeros e observar a resposta em frequência, uma vez dentro do especificado, a equação de diferenças é calculada.

O Método da “Transformação Bilinear Z” consiste em uma mudança de variável, onde a função de transferência de um filtros analógico é convertida em digital.

## 4.2 - Projeto de Filtros IIR com inserção de Polos e Zeros

O método de inserção de polos e zeros é calculado nestas etapas:

1. Inserir polos e zeros até que os sistema atinja as descrições e características desejadas.
2. Expandir os polos na equação 38 até encontrar a Função de Transferência  $H(z)$ .
3. Calcular a equação de diferenças. (Detalhado na seção 2.3.6).
4. Filtrar. (Detalhado na seção 4.4).

O seguinte exemplo demonstra como utilizar este método para criar um filtro *notch*. O exemplo também mostra os efeitos dos polos e dos zeros no círculo unitário.

### 4.2.1 - Exemplo da Utilização do método de inserção de polos e zeros para criação de um filtro *Notch*.

Retirado de (Ifeachor e Jervis, 1993, p. 381), este exemplo cria uma filtro *notch*, usando as técnicas de inserção de polos e zeros com as seguintes características:

- Frequência rejeitada: 50 Hz
- Atenuação de 3 dB:  $\pm 5$  Hz
- Frequência de amostragem: 500 Hz

Antes de começar a inserir pontos sobre o círculo unitário, imagina-se que cada polo/zero inserido gera um efeito sobre a resposta em frequência. O “zero” de zerar a resposta em frequência.

O gráfico 17 mostra diferentes posições e os efeitos sobre a resposta:

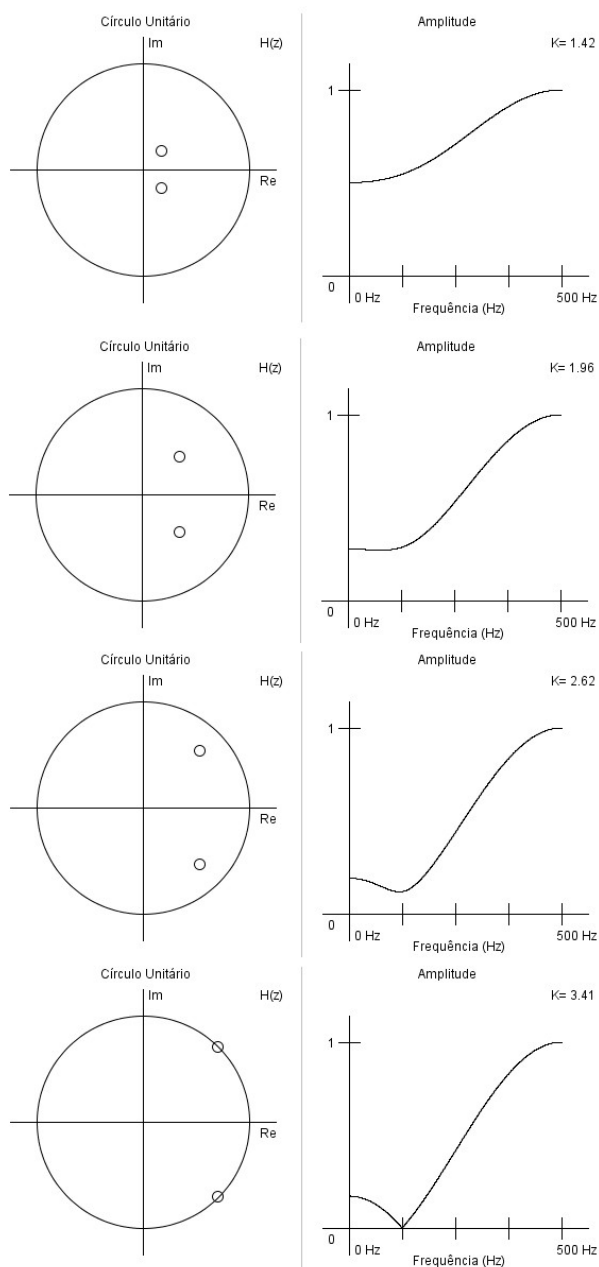
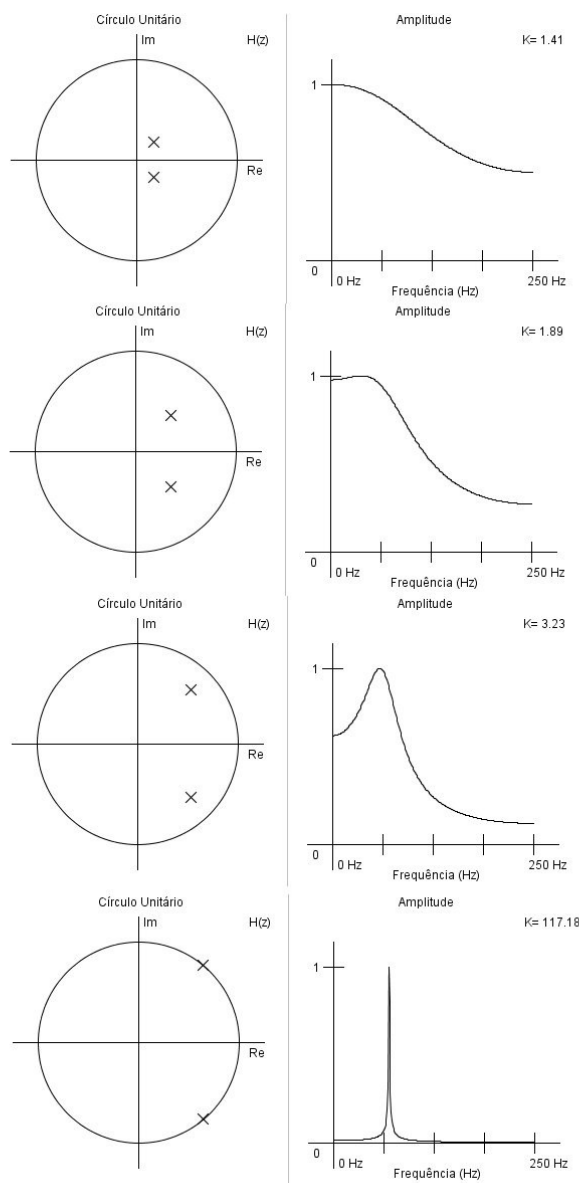


Figura 17 - Gráfico do plano Z com zeros em diferentes posições.

Assim, quando deseja-se filtrar uma frequência específica, é desejável que o zero fique sobre o círculo unitário, na posição que se refere a frequência a se rejeitar. Por exemplo, se  $\pi$  (meia volta) equivale a metade frequência de amostragem, então para encontrar um frequência específica, basta realizar uma regra de três.

Já os polos realizam um comportamento conforme a Figura 18:



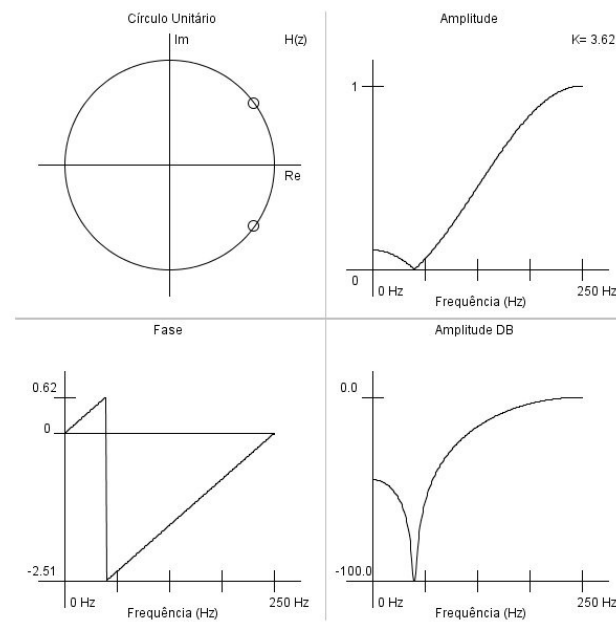
*Figura 18 - Gráfico do plano Z com polos em diferentes posições.*

Assim, juntando polos e zeros em um diagrama, pode-se projetar o filtro desejado. Como a frequência que deseja-se eliminar é 50 Hz, faz-se uma regra de três:

$$\frac{\pi}{X} = \frac{250}{50}$$

$$X = \frac{\pi}{5}$$

Logo, sabe-se que um zero ao menos tem que ser inserido sobre os  $36^\circ$ . Fazendo o mesmo, a resposta em frequência ficaria conforme Figura 19:



*Figura 19 - Gráfico do plano Z contendo um zero com coordenadas polar igual a 1 em  $36^\circ$ , e seu conjugado.*

Porém, observando a Figura 19, as bandas de transições estão se estendendo demasiadamente. Para isso, insere-se um polo no mesmo ângulo, porém, um pouco atrás do zero. O raio apropriado do polo, segundo Ifeachor e Jervis (1993, p. 382), é de 0,937. Desta maneira sua interferência não afetará a característica do zero de anular a frequência desejada, mas irá estreitar a banda de transição. A Figura 20 mostra o sistema completo e sua resposta de frequência:

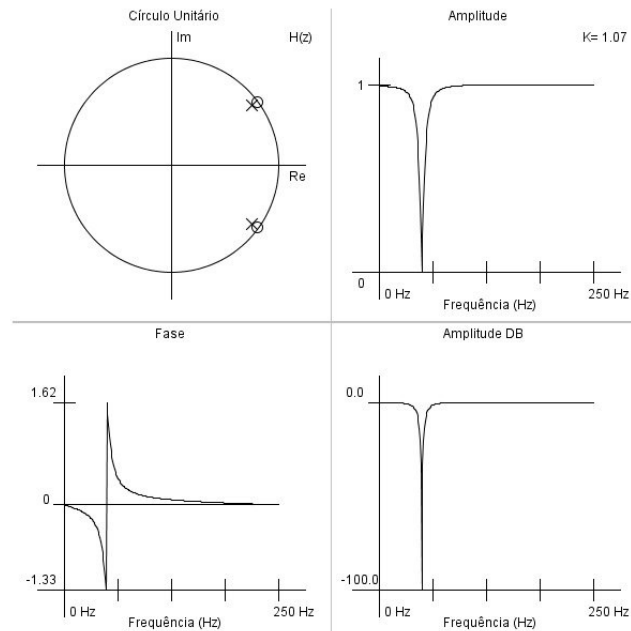


Figura 20 - Gráfico do plano Z contendo um zero com coordenadas polar igual a 1 em 36 graus, e seu conjugado junto de um polo com coordenadas polares igual a 0,937 em 36 graus e seu conjugado.

Tendo em mãos os polos e zeros do sistema, inicia-se o cálculo da equação de diferenças. A equação de diferenças deste sistema é:

$$y(n) = x(n) - 1,6180x(n-1) + x(n-2) + 1,5161y(n-1) - 0,8780y(n-2) \quad 39$$

Finalmente, para se aplicar este filtro em um sinal discretizado, basta implementar uma rotina em tempo real que aplique a equação 39 sobre o sinal e a resposta será um sinal filtrado.

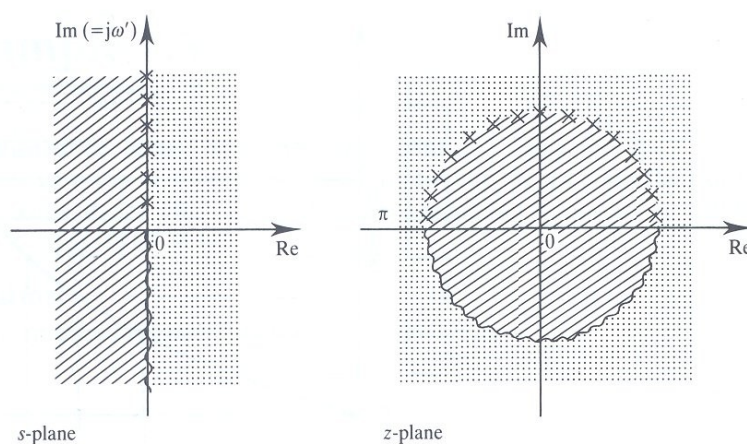
### 4.3 - Projetos de Filtros IIR com a Transformação Bilinear Z (BZT)

Este método consiste em criar uma aproximação digital de um filtro analógico. Existe diversos métodos para isto, descritos por Iffachor e Jervis em seu livro, *Digital Signal Processing*. Aqui será descrita apenas a BZT pela sua praticidade em transformar funções de transferência do domínio S para o Z.

A BZT consiste praticamente em fazer a seguinte substituição na função de transferência do plano S:

$$s = k \frac{z-1}{z+1}, k = \frac{2}{T}$$

Esta transformação mapeia a função de transferência,  $H(s)$ , do plano S criando uma versão analógica do sistema no plano Z. A Figura 21 mostra como o mapeamento funciona:



*Figura 21 - Mapeamento do Plano S no Plano Z.*

*Fonte: Adaptado de IFEACHOR (1993).*

Porém, este mapeamento causa um efeito colateral, o *warping effect*. Este efeito, pode ser observado traçando a resposta em frequência do filtro analógico contra a do filtro digital, conforme pode ser observado na Figura 22:

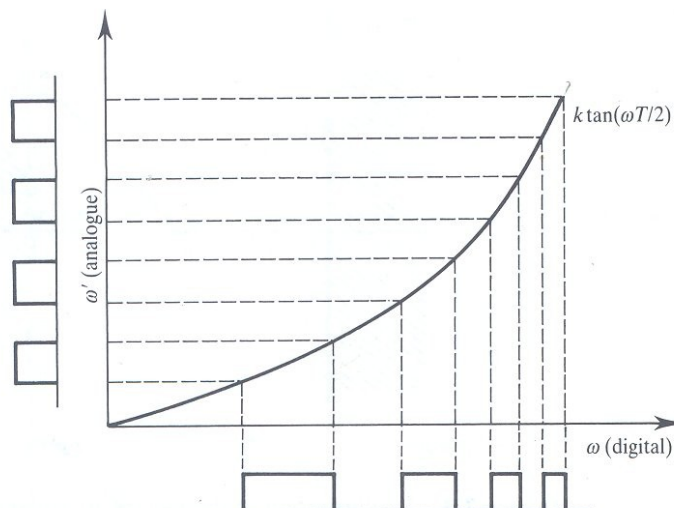


Figura 22 - Gráfico do Efeito Warp.

Fonte: Adaptado de IFEACHOR (1993).

Nota-se que a transformação não é linear, então o resultado da transformação cria um filtro próximo da sua versão analógica e não idêntica. Esse efeito pode ser compensado realizando um *pré-warp* antes da transformação, porém, este efeito pode ser ignorado se a resposta da transformação resultar em um filtro que esteja de acordo com a especificações. Esse efeito deve ser considerado em casos, por exemplo, onde um filtro analógico será substituído por um digital transformado com a BZT.

#### 4.4 - Utilizando a Equação de Diferenças

Este tópico traz um exemplo de como aplicar a equação de diferenças em uma rotina.

Supondo o seguinte sinal:

$$x = [1 \ 2 \ 3 \ 2 \ 1]$$

E para este exemplo será tomada a equação 39, da página 41, para aplicação, logo, o sinal  $x$  será aplicado nesta equação. Como pode-se ver na equação, ele depende do valor atual de  $x$ , alguns valores passados de  $x$  e alguns valores passados de  $y$ , que é a saída. Então criaremos uma tabela, inicializada com zero, para armazenar estes valores.



$$\begin{aligned}
 x0 &= 0 \\
 x1 &= 0 \\
 x2 &= 0 \\
 y1 &= 0 \\
 y2 &= 0
 \end{aligned}$$

Então, como sinal x possui 5 posições, o a aplicação ocorre em cinco etapas.

Etapas 1:  $n = 1$

$$\begin{aligned}
 x0 &= 1 \\
 x1 &= 0 \\
 x2 &= 0 \\
 y1 &= 0 \\
 y2 &= 0
 \end{aligned}$$

E aplicando na fórmula:

$$\begin{aligned}
 y(1) &= 1 - 1,6180 * 0 + 0 + 1,5161 * 0 - 0,8780 * 0 \\
 y(1) &= 1
 \end{aligned}$$

Etapas 2:  $n = 2$

No começo de cada etapa, é necessário deslocar os valores de x e de y em uma posição, para que fiquem armazenados. Este processo não era visível na etapa 1, pois todos os valores eram iguais a zero. Então fazendo este deslocamento e já inserindo o novo valor de x1:

$$\begin{aligned}
 x0 &= 2 \\
 x1 &= 1 \\
 x2 &= 0 \\
 y1 &= 1 \\
 y2 &= 0
 \end{aligned}$$

Aplicando a fórmula:

$$\begin{aligned}
 y(2) &= 2 - 1,6180 * 1 + 0 + 1,5161 * 1 - 0,8780 * 0 \\
 y(2) &= 1,8981
 \end{aligned}$$

Etapas 3:  $n = 3$

$$\begin{aligned}
 x0 &= 3 \\
 x1 &= 2 \\
 x2 &= 1 \\
 y1 &= 1,8981 \\
 y2 &= 1
 \end{aligned}$$

Aplicando a fórmula:

$$y(3) = 3 - 1,6180 * 2 + 1 + 1,5161 * 1,8981 - 0,8780 * 1$$

$$y(3) = 2,7637$$

Etapa 4: n = 4

$$x0 = 2$$

$$x1 = 3$$

$$x2 = 2$$

$$y1 = 2,7637$$

$$y2 = 1,8981$$

Aplicando a fórmula:

$$y(4) = 2 - 1,6180 * 3 + 2 + 1,5161 * 2,7637 - 0,8780 * 1,8981$$

$$y(4) = 1,6695$$

Etapa 5: n = 5

$$x0 = 1$$

$$x1 = 2$$

$$x2 = 3$$

$$y1 = 1,6695$$

$$y2 = 2,7637$$

Aplicando a fórmula:

$$y(5) = 1 - 1,6180 * 2 + 3 + 1,5161 * 1,6695 - 0,8780 * 2,7637$$

$$y(5) = 0,8686$$

Terminando as cinco etapas temos:

$$y = [1 \ 1,8981 \ 2,7637 \ 1,6695 \ 0,8686]$$

Este processo explica como funciona a aplicação da equação de diferenças.

## 5 - MATERIAIS E MÉTODOS

Neste tópico, serão descritas as ferramentas utilizadas para o desenvolvimento do projeto, desde *softwares* de apoio até os *software* de desenvolvimento e apresentação.

### 5.1 - Sistema de Processamento de Sinais Biomédicos

A Plataforma Didática de Ensino em Engenharia Biomédica é formada por um conjunto de ferramentas para auxiliar o ensino na área de Engenharia Biomédica. Ela foi proposta inicialmente por Andrighetto, Possa, Rathke e Adur, 2008.

Segundo Possa (2008), apenas 7,5% dos cursos de Engenharia Elétrica (e afins) possuem disciplinas relacionadas a Engenharia Biomédica. Estes 7,5%, que correspondem a apenas 16 cursos de formação de Engenharia Elétrica, geram um total de 2.208 horas-aulas em disciplinas referentes a Engenharia Biomédica. Deste total, apenas 310 horas-aulas são dedicadas à prática.

Sendo assim, segundo Andrighetto, 2008, “O objetivo da plataforma é criar um instrumento de auxílio ao ensino e aprendizagem dos temas técnicos e práticos que formam o engenheiro biomédico, principalmente quanto a prática.”

### 5.2 - Softwares Livres

A Plataforma Didática de Ensino em Engenharia Biomédica toma uma iniciativa de valorizar e patrocinar o uso de *softwares* livres como alternativa a muitos *softwares* comerciais. Este tópico trará um relato sobre alguns que foram utilizados neste projeto.

#### 5.2.1 - JUDE/Community e UML

O JUDE/Community é um *software* livre, implementado em JAVA, utilizado para criação e edição de diagramas UML. A versão gratuita deste *software* permite a criação de diagramas de UML na versão 1.4 e com suporte a algumas ferramentas da versão 2.0. (<http://jude.change-vision.com>)

O UML – Unified Modeling Language (Linguagem de Modelagem Unificada) é uma linguagem que indica as formas como um *software* pode ser representado em diversas etapas de seu desenvolvimento. (E. Medeiros, 2004). Uma vez unificada a maneira de se projetar um *software* específico, ele independe da linguagem de programação que será implementado.

Isto torna possível uma visão única dos objetivos, porém diversificada no meio em que será

desenvolvido, trazendo a flexibilidade ao programador. Um projeto unificado e bem representado serve como uma documentação completa do que se desenvolve, possibilitando uma previsão das funções do sistema quando completo e em caso de modificações, torna-se muito mais simples visualizar os eventos em forma de diagramas do que em forma de código fonte.

### 5.2.2 - Desenvolvimento em JAVA

Até hoje, a plataforma Java já atraiu mais de 5 milhões de desenvolvedores de *software*. Ela é usada em todos os principais setores e está presente em uma ampla gama de dispositivos, computadores e redes. (JAVA, 26-11-2008).

JAVA é uma linguagem de programação que prioriza a facilidade na programação e criação de *softwares*, independente da cobrança de direitos autorais. Com a ampla utilização da linguagem JAVA nos diversos *softwares* que operam hoje no mercado, há a necessidade básica de se possuir a máquina virtual JAVA instalada na maioria dos computadores que estão espalhados pelo mundo. Em consequência, a máquina virtual JAVA torna possível a implementação de sistemas utilizando java com a segurança de que o computador do usuário final vai estar pronto para utilizar o produto.

A tecnologia Java está presente em computadores, celulares e outros dispositivos portáteis e outros produtos especializados e permite diversas facilidades, como:

- Criar um *software* em uma plataforma e executá-lo em praticamente qualquer outra;
- Criar programas para execução em navegadores e serviços da Web;
- Desenvolver aplicativos no lado do servidor usados tanto em fóruns, lojas e pesquisas on-line, como no processamento de formulários HTML;
- Combinar aplicativos ou serviços usando a linguagem Java para criar aplicativos ou serviços altamente personalizados;
- Criar aplicativos potentes e eficazes para telefones celulares, processadores remotos, produtos de consumo de baixo custo e praticamente qualquer outro dispositivo com tecnologia digital.

### 5.2.3 - Java Applets

*Applet* é um *software* que é executado ou interpretado (e/ou apresentado) por outro programa, neste caso específico, telas gráficas ou informativas que são implementadas em JAVA e são apresentadas em um navegador Web (Internet Explorer, Firefox etc.). A única restrição neste caso é a necessidade do navegador suportar a Máquina Virtual JAVA.

JAVA Applet é uma ferramenta interessante para a criação de sistemas onde dados são processados, pois a Plataforma Virtual JAVA permite a exibição de gráficos integrados com o navegador, ou seja, estes dados calculados podem ser exibidos em formas gráficas, tudo isso sem a necessidade da execução um *software* exclusivo. JAVA Applet também permite a entrada de dados, onde o usuário do Applet pode entrar com dados, executar e visualizar de uma maneira gráfica, sem que precise interpretar números e imaginar o comportamento da saída.

J.M. Kootsey, G. McAuley e H. Liu (2001) descrevem a utilização de JAVA Applets como uma tecnologia excelente para implementar simulações para ensino, pois os Applets podem facilmente ser combinados com textos, figuras, animações e vídeo.

F. Saharil, A.M. Hasbi e H. Hussin (2004) afirmam que Java Applets integrados a textos HTML atuam como uma excelente combinação para criação de Web-site educacionais interativos.

#### 5.2.3.1 - Componentes JAVA

Existem uma série de bibliotecas que podem ser utilizadas para a criação de Applets. Uma delas é a *Abstract Window Toolkit* (AWT), que possui diversos componentes para construção de janelas, *combobox*, áreas de texto, barra de rolagem, área de imagem, etc.

Outra biblioteca é a *Swing*, que é uma versão mais atualizada e sofisticada do AWT. Neste trabalho, é utilizada uma integração entre as duas ferramentas, pois a sofisticação da *Swing* não permite certos desenhos em tempo real. Porém, a *Swing* não foi excluída totalmente, pois algumas outras ferramentas tornam o *Applet* com o *layout* mais agradável.

As ferramentas mais utilizadas do AWT e Swing são botões, painéis, tabelas, barra de rolagem, bordas, campo de inserção de textos, áreas de textos e combo-box.

#### 5.2.4 - Scilab

Scilab é um *software* científico para computação numérica com um poderoso ambiente aberto para engenharia e aplicações científica. Também conhecido por ser a solução livre de direitos autorais do Matlab. (<http://www.scilab.org>)

Desde 1994, o Scilab é distribuído gratuitamente com seu código fonte na Internet. Atualmente é utilizado em ambientes educacionais e industriais em todo o mundo.

Scilab inclui centenas de funções matemáticas com a possibilidade de adicionar interatividade com programas de diversas linguagens (C, C++, Fortran etc.).

### 5.3 - Validação das Ferramentas dos Tutoriais

A validação é o processo que garante a funcionalidade e efetividade das funções desenvolvidas. Como as funções desenvolvidas abordam sinais, a ferramenta utilizada para garantir a funcionalidade dos filtros e funções foi a Transformada de Fourier, que pode ajudar a garantir que os filtros estejam funcionando adequadamente.

#### 5.3.1 - Transformada de Fourier

A Transformada de Fourier é uma variação da Transformada Z, que nos permite analisar as componentes senoidais de um sinal. Uma vez que aplicamos a Transformada de Fourier em um sinal, podemos saber quais são os senos que compõem este sinal e a amplitude de cada um deles.

Assim, para saber se um filtro está funcionando adequadamente ou não, basta realizar uma transformada de Fourier antes e uma depois, desta maneira pode-se observar se no segunda caso as componentes que filtramos desapareceram.

### 5.4 - Avaliação dos Tutoriais

A avaliação consiste na utilização de algum método para garantir a efetividade de todo o trabalho desenvolvido. Então, tratando-se de sistemas Web, para a avaliação foi escolhido o *WebMac Senior (4.0)* (SMALL e ARNONE, 1999), que consiste em um formulário aplicado aos usuários. O resultado é analisado e com ele pode-se tirar alguma conclusão sobre a efetividade do trabalho.

#### 5.4.1 - Webmac Senior (4.0)<sup>©</sup>

A avaliação utilizando o *Webmac Senior* consiste na aplicação do formulário proposto a um grupo de usuários do produto, focado em quatro atributos. Nele, mede-se o grau de concordância entre a questão abordada e a experiência individual do avaliador na utilização do produto, através de 32 questões (SMALL e ARNONE, 1999).

Numericamente, cada item é quantificado da seguinte maneira:

- 3 = concordo plenamente;
- 2 = concordo parcialmente;
- 1 = discordo em partes;
- 0 = discordo totalmente;

- **NA** = não aplicável.

Para cada item NA, deve-se atribuir um dos seguintes valores:

- **0** = o sistema avaliado seria beneficiado se tivesse incluído esse item;
- **1** = o sistema avaliado não requer este item;
- **2** = o sistema avaliado ficou melhor por não incluir este item.

#### 5.4.2 - Pontuação do WebMac Senior

Quando todas as questões estiverem preenchidas com um valor numérico, as 32 questões são arranjadas em quatro grupos, cada grupo representando um atributo:

- **E** para o quão estimulante o produto é;
- **S** para o quão significativo o produto é;
- **O** para o quão organizado o produto é;
- **F** para o quão fácil-de-usar o produto é.

<b><u>E</u></b>	<b><u>S</u></b>	<b><u>O</u></b>	<b><u>F</u></b>
1.____	2.____	3.____	4.____
5.____	6.____	7.____	8.____
9.____	10.____	11.____	12.____
13.____	14.____	15.____	16.____
17.____	18.____	19.____	20.____
21.____	22.____	23.____	24.____
25.____	26.____	27.____	28.____
29.____	30.____	31.____	32.____
Total E ____	Total S ____	Total O ____	Total F ____

Posteriormente, o total de cada usuário (individualmente) é plotado em um gráfico conforme Figura 23.

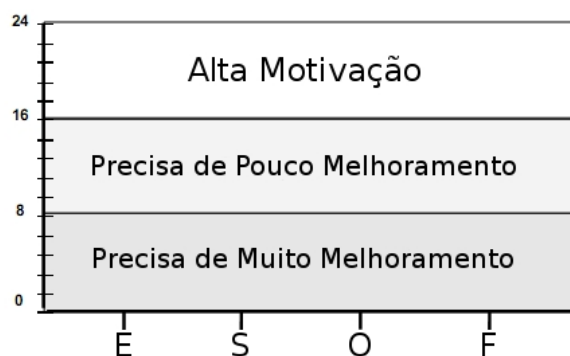


Figura 23 - Exemplo do Gráfico E.S.O.F.

Fonte: Adaptado de SMALL e ARNONE (1999).

Onde o valor de E, S, O e F são desenhados em sua posição no eixo Y conforme o valor resultante no eixo X. Conforme a área em que o ponto se encontrar, o resultado é uma das três afirmações:

- Alta Motivação;
- Precisa de Pouco Melhoramento;
- Precisa de Muito Melhoramento.

#### 5.4.3 - Classificação e Avaliação do Produto

Depois que o gráfico E.S.O.F. estiver desenhado, pode-se ter uma prévia das falhas do sistema. Por exemplo, em qual ponto exato está havendo falha, baseado nos quatro grupos E.S.O.F.

Depois de analisado o gráfico E.S.O.F., para a avaliação do produto, os resultados individuais são somados conforme as expressões 40 e 41.

$$V = E + S \quad 40$$

$$XS = O + F \quad 41$$

A pontuação V reflete o quão valioso o produto é e o XS o quão motivante o *site* é, ou seja, a sua expectativa de sucesso. Por fim, todos os valores individuais podem ser plotados em um único gráfico, conforme a Figura 24.



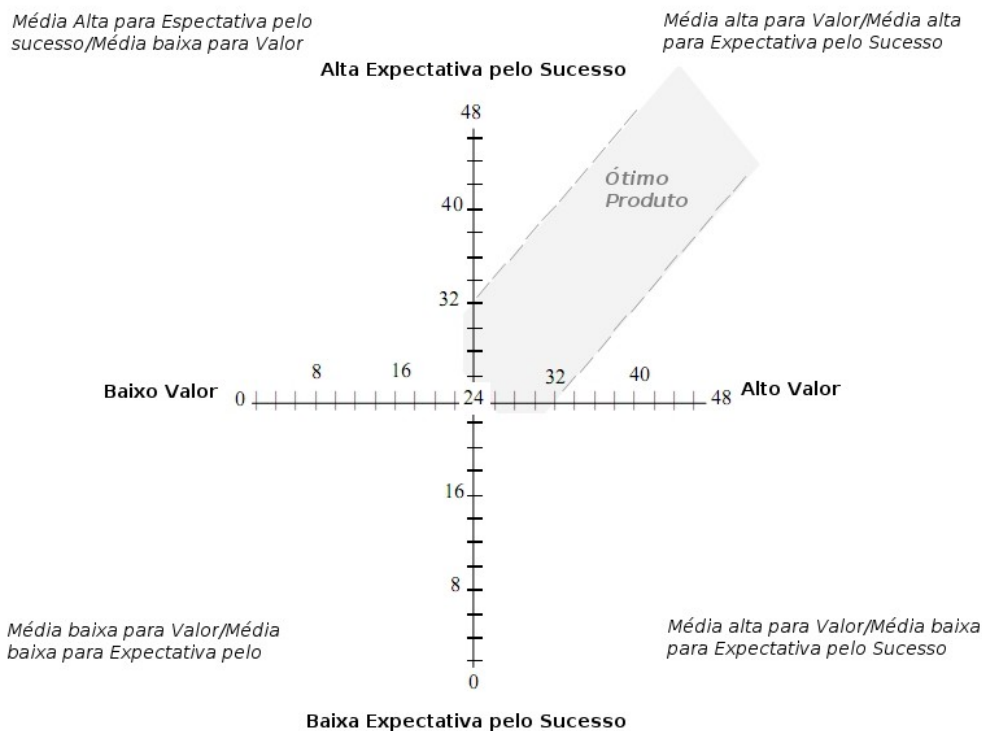


Figura 24 - Gráfico VXS.

Fonte: Adaptado de SMALL e ARNONE (1999).

Por fim, se os pontos ou uma grande quantidade dos pontos individuais se localizarem dentro da área em cinza, significa que o produto é um “Ótimo Produto”, avaliando-o positivamente.

#### 5.4.4 - Questionário Aplicado

O questionário foi desenvolvido focado nas 32 questões propostas pelo *Webmac Senior 4.0*, salvo algumas modificações para especificar atributos do produto desenvolvido. Este formulário pode ser encontrado no ANEXO I – QUESTIONÁRIO WEBMAC 4.0 SENIOR.

## 6 - DESENVOLVIMENTO

Esta seção descreve as análises iniciais do projeto, a utilização das ferramentas e o desenvolvimento do projeto.

### 6.1 - Projeto

Como visto, processamento Digital de Sinais tem se destacado como uma área complexa e que requer um alto conhecimento de ferramentas matemáticas para o seu projeto. Sendo assim, serão desenvolvidos tutoriais de ensino para auxiliar no ensino de filtragem digital de sinais, com um foco neste trabalho em sinais biomédicos.

Partindo do pressuposto de que o trabalho desenvolvido deveria ser um sistema capaz de ensinar alunos sobre filtragem digital aplicado a Engenharia Biomédica, optou-se por dar continuidade ao trabalho de Adur (2008), Rathke (2008), Possa (2008) e Andrighetto (2008) e criar junto ao Portal Saúde+Educação<sup>TM</sup> um sistema tutorial capaz de ensinar esse conteúdo na forma de uma aula virtual.

Neste tópico são abordadas passo-a-passo as etapas de desenvolvimento do projeto, desde a escolha dos conteúdos abordados, requisitos, validação e avaliação.

#### 6.1.1 - Conteúdo Abordado

A escolha do conteúdo foi dada de acordo com os seguintes critérios:

- O aluno deve aprender na teoria e na prática (Requisitos mínimos dos conteúdos do Portal Saúde+Educação<sup>TM</sup>);
- O aluno deve terminar as aulas práticas tendo o conhecimento de como se projetam filtros digitais;
- O aluno deve terminar as aulas prática tendo o conhecimento de como filtros digitais devem ser utilizados para que o sinal de eletroencefalograma mantenha-se íntegro.

Assim, com estes critérios os assuntos foram escolhidos, conforme os seguintes tópicos:

1. Sinais e sistemas no tempo discreto: o aluno precisa conhecer como um sistema discreto funciona e como é formado um sinal discreto. É o mínimo para trabalhar com filtragem digital, pois um filtro digital é um sistema discreto.

2. Transformada Z: é a ferramenta básica em sistemas discretos. Ela é que descreve a transformação de um sistema ou sinal do tempo discreto para o tempo contínuo e vice-versa.
3. Filtros Digitais: são um tipo específico de sistema discreto. Ele é o foco desta abordagem e deve ser analisado.
4. Filtro FIR (Método Janelas): são os filtros de resposta finita ao impulso e o método janela, que é um método de se projetar filtros FIR, foi escolhido por ser o mais utilizado.
5. Filtro IIR (Inserção de Polos e Transformação Bilinear Z): Filtros IIR são os filtros de resposta ao impulso infinito e o método de inserção foi escolhido para facilitar o ensino e o da Transformação Bilinear Z por ser o método mais utilizado.
6. Aplicação de filtros a eletroencefalograma: como aplicar os filtros é fundamental, sendo que os métodos da aplicação de um filtro FIR e IIR são diferentes. Também envolve neste tópico as aplicações de filtros digitais em sinais biomédicos (EEG), como se deve e não se deve aplicar um filtro, de maneira a não danificar um sinal.

Estes são os temas, em ordem de prioridades, em que um aluno precisa ter conhecimento para projetar e aplicar um filtro digital.

Porém, antes disto, o aluno precisa ter o conhecimento de sinais e sistemas no tempo contínuo. Por sua vez, este conteúdo não foi abordado neste trabalho pois, se tivesse sido abordado, tornaria o conteúdo do trabalho extenso demais.

Sabendo o conteúdo que precisa ser abordado, foram escolhidas as ferramentas.

### 6.1.2 - Estruturação do Trabalho

Este tópico descreve como o trabalho foi arquitetado. Uma vez que os tutoriais serão imersos no Portal Saúde+Educação<sup>TM</sup>, eles precisam respeitar os requisitos deste sistema.

Os tutoriais do Portal Saúde+Educação<sup>TM</sup> são divididos em um conjunto de tutoriais. Cada tutorial possui um conjunto de tópicos, cada tópico possui os seguintes sub-tópicos:

- Conteúdo;
- História;
- Correlação Clínica;

- Mão na Massa;
- Auto Avaliação.

E ainda cada sub-tópico pode ser dividido em títulos numerados. Sendo assim, vindo dos trabalhos anteriores, o Portal Saúde+Educação<sup>TM</sup> já contém um módulo chamado “Sistema de Processamento de Sinais Biomédicos”, no qual este trabalho será inserido junto dos trabalhos já desenvolvidos nesta área.

Dentro do módulo “Sistema de Processamento de Sinais Biomédicos” existem dois tópicos decorrentes de trabalhos anteriores. A proposta deste tutorial é a criação de um novo tópico, inserido neste tutorial para abordar filtragem digital de Sinais junto do trabalho de Briese (2009), que aborda o conteúdo de filtros analógicos. Então, será criado um tópico chamado “Filtragem de Sinais”, contendo os cinco sub-tópicos “Conteúdo, História, Correlação Clínica, Mão na Massa e Auto Avaliação”.

Pelo fato do Portal Saúde+Educação<sup>TM</sup> se tratar de um sistema *web*, os tutoriais devem ser redigidos em formato HTML. Por ser um sistema *web* e suportar a Máquina Virtual JAVA, também serão utilizados para dar apoio ao “Conteúdo” JAVA Applets.

Para “Mão na Massa”, por requisito do Portal Saúde+Educação<sup>TM</sup>, há a necessidade de tutoriais práticos. Como o escopo deste trabalho não previu implementação em *hardware* dos filtros, foi optado pela utilização do *software* Scilab para realização das práticas sobre filtragem digital.

Por fim, com todas as ferramentas implementadas, redigidas e prontas para publicação, foram inseridas no site do Portal Saúde+Educação<sup>TM</sup>, no sítio [www.saudeeducacao.ufsc.br](http://www.saudeeducacao.ufsc.br).

### 6.1.3 - Modelagem

Para melhorar o entendimento, o escopo da modelagem está focado apenas no conteúdos dos Applets e não nos tutoriais em geral. Esta maneira foi escolhida para que futuramente se torne clara a diferença entre o que é implementado e o que é escrito.

O primeiro passo da modelagem é descobrir quem são os atores do sistema e em que macro sistema eles estão inseridos. Neste caso, o “Ator Aluno” utiliza o sistema e o “Ator Sistema” efetua os cálculos e ambos os atores estão inseridos no sistema “Aprender Filtragem Digital de Sinais”, que será chamando de Visão Geral.

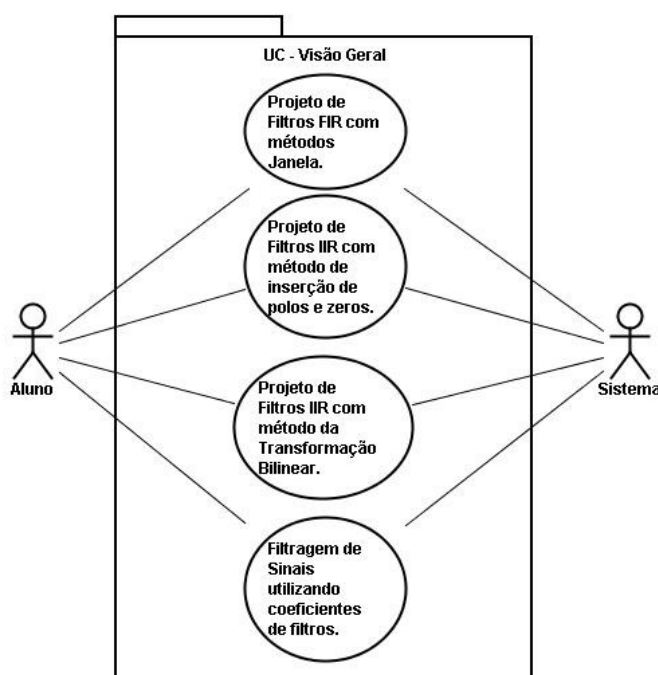
A Visão Geral do sistema mostra quais sistemas independentes os atores terão acesso. Para que isto seja possível, o conteúdo precisa ser explorado e descobrir em que pontos há a necessidade de uma ferramenta Applet para apoiar o conteúdo.

Visando que o Applet deve ser uma ferramenta prática, foram observados os seguintes pontos:

- Momento em que o filtro é criado;
- Momento em que o sinal é filtrado.

Para o “Momento em que o filtro é criado”, são encontrados dois tipos de filtros, os filtros FIR e os filtros IIR. Para filtros FIR, será necessário abordar um método de criação de filtros. Para IIR, são necessários abordar dois métodos de criação. Já o “Momento em que o sinal é filtrado”, é necessário abordar os filtros em funcionamento.

Assim, a Visão Geral do sistema se dá conforme a Figura 25.



*Figura 25 - Visão Geral do sistema.*

Na Figura 25, pode-se observar quatro macro-sistemas em que tanto o Aluno quanto o Sistema atuam. O Aluno atua diretamente na entrada e saída de dados, ao passo que o Sistema processa os dados de entrada, transformando-os em uma saída.

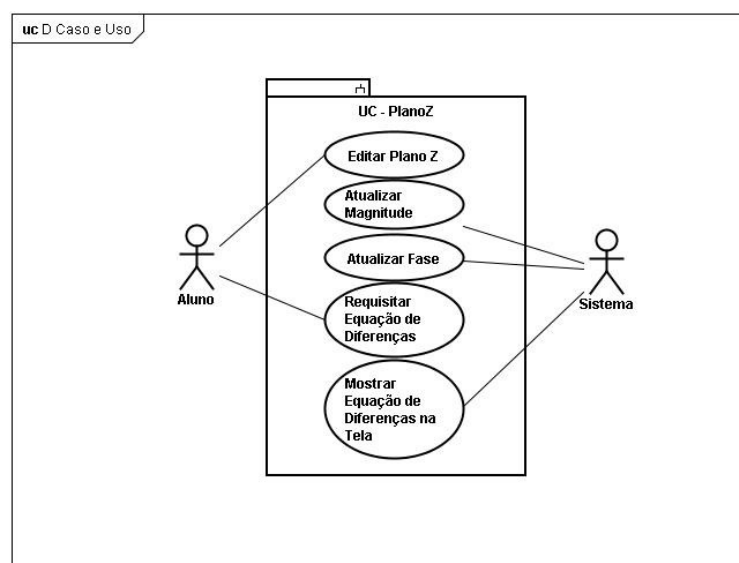
Os quatro casos da Figura 25 representam quatro sistemas diferentes que precisam ser implementados para dar apoio aos tutoriais. Estes casos são descritos detalhadamente a seguir.

#### 6.1.3.1 - Projeto de Filtros IIR com método de inserção de polos e zeros

O “projeto de filtros IIR com método de inserção de polos e zeros” é também chamado, no projeto, de PlanoZ.

De acordo com o método de inserção de polos e zeros, sabe-se que o Aluno deve entrar com as posições do polos e zeros no plano Z, e em seguida, visualizar o efeito daquele sistema sobre os sinais, em fase e amplitude. Neste caso, o ator Sistema é responsável por efetuar os cálculos e mostrar para o Aluno gráficos destes comportamentos.

Sendo assim, o diagrama de Caso de Uso do PlanoZ se dá conforme a Figura 26.



*Figura 26 - Diagrama de Caso de Uso do PlanoZ.*

No diagrama de Caso de Uso do PlanoZ podemos observar as ações que são executadas pelo Aluno e pelo Sistema. Neste caso, o Aluno pode editar o plano Z, que consiste em adicionar, remover ou editar os polos do sistema discreto. Por fim, o aluno pode requisitar a equação de diferenças para que possa utilizar em seus projetos o filtro projetado.

Por outra parte, o Sistema é responsável por atualizar a magnitude e a fase no gráfico em que o Aluno visualiza. Ele também é responsável por processar a equação de diferenças e mostrar para o aluno.

Com o Caso de Uso da Figura 26, pode-se saber tudo que o PlanoZ será capaz de fazer, porém não como. Este é o limite do diagrama de Caso de Uso. Para mais detalhes, foi criado o diagrama de atividades que demonstra como o sistema funciona em atividades e o diagrama de classe, que mostra como se distribui as classes para a implementação final. Estes dois diagramas podem ser encontrados no APÊNDICE I – DIAGRAMAS DE MODELAGEM, como “Diagrama de Atividades do PlanoZ”, “Diagrama de Classe do PlanoZ”, “Diagrama de Atividades do Calcula Magnitude” e “Diagrama de Atividades do Calcula Fase”. As atividades simples como “Botão Reset”, “Retorna” etc não são detalhadas pela simplicidade da função.

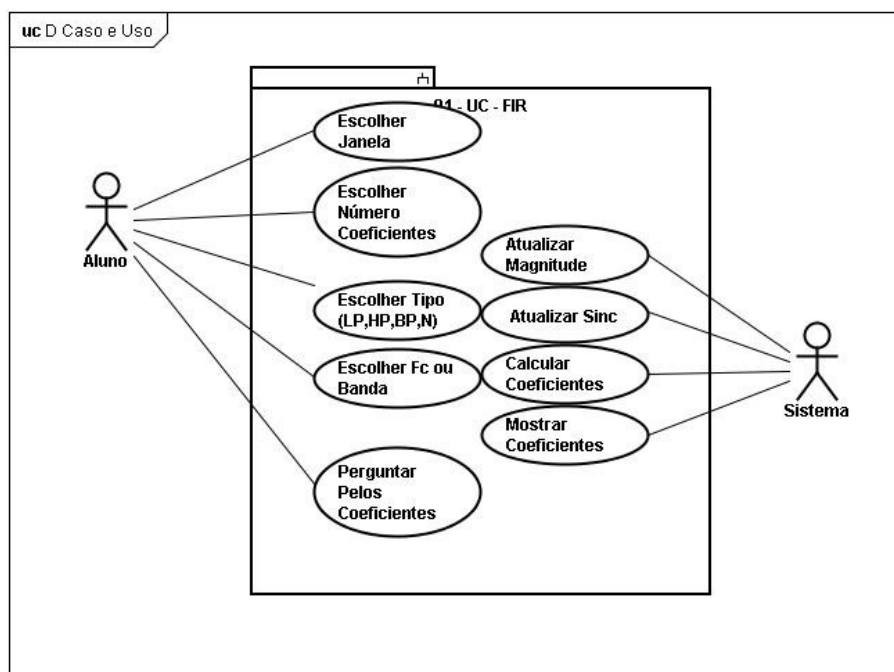
Com estas informações, tem-se conhecimento do que o sistema será capaz de realizar, quem irá atuar, como irá funcionar e a base de como será implementado, independentemente da linguagem escolhida.

#### 6.1.3.2 - Projeto de Filtros FIR com métodos Janela

O “Projeto de Filtros FIR com métodos Janela” é chamado, no projeto, de FiltroFIR.

De acordo com o método de janelas de projeto de filtros FIR, o Aluno deve entrar com um conjunto de valores requeridos e em seguida visualizar o efeito daquele sistema sobre os sinais em amplitude. Neste caso, o ator Sistema é responsável por efetuar os cálculos e mostrar para o Aluno gráficos destes comportamentos.

Sendo assim, o diagrama de Caso de Uso do FiltroFIR se dá conforme a Figura 27.



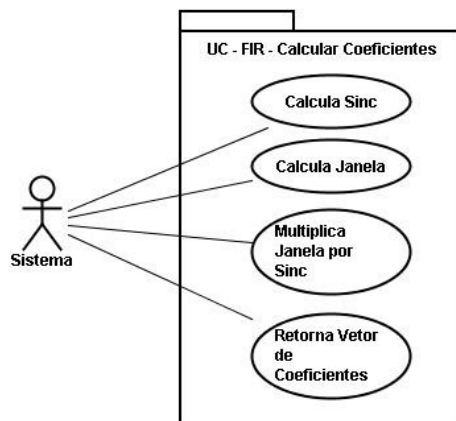
*Figura 27 - Diagrama de Caso de Uso do FiltroFIR.*

No diagrama de Caso de Uso do FiltroFIR, podemos observar as ações que são executadas pelo Aluno e pelo Sistema. Neste caso, o Aluno pode projetar o filtro com diversas variáveis e visualizar os seus resultados. Por fim, o Aluno pode requisitar os coeficientes para que possa utilizar em seus projetos o filtro criado.

O Sistema tem o papel de calcular e mostrar para o Aluno o que acontece com os dados que Aluno informou.

Porém, na Figura 28, no Caso: “Calcular Coeficientes”, foi identificado que o caso também precisa ser analisado independentemente, então, este caso gerou um outro caso de uso que mostra com mais detalhes o funcionamento deste processo. O Caso de Uso pode ser visualizado na Figura 28.





*Figura 28 - Diagrama de Caso de Uso do Caso "Calcular Coeficientes".*

Neste Caso, temos apenas o Sistema como ator, uma vez que o Aluno não está presente nas atividades de cálculo.

Por fim, com todos os casos do FiltroFIR observado, encontra-se no APÊNDICE I – DIAGRAMAS DE MODELAGEM os Diagramas de Atividade e Diagramas de Classe do sistema. As classes que já foram detalhadas nas seções anteriores não são reescritas no projeto, pois como cada fase do projeto necessita da anterior, as classes são compartilhadas e não há a necessidade de refazer esta modelagem.

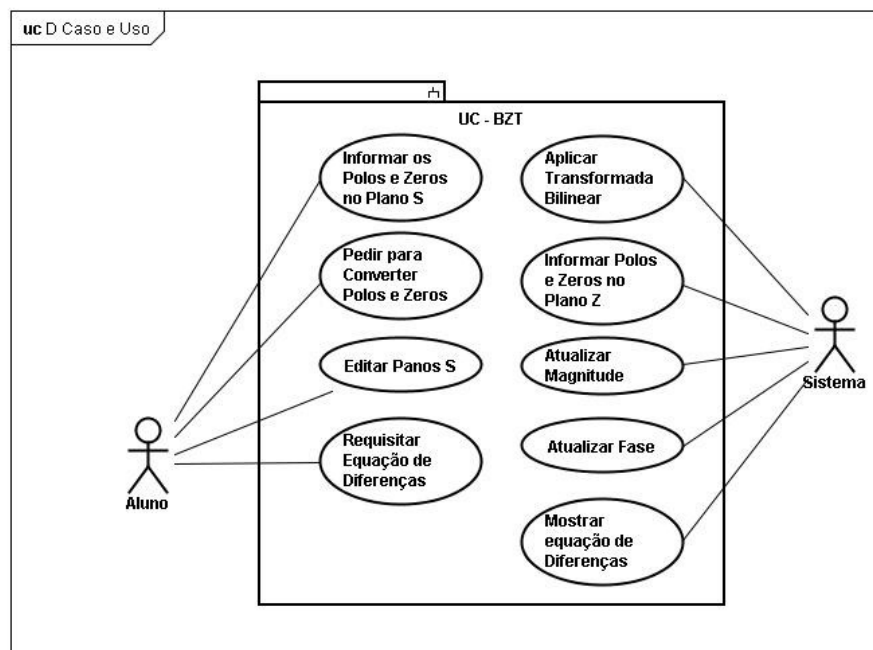
Com estas informações, tem-se conhecimento do que o Sistema será capaz de realizar, quem irá atuar, como irá funcionar e a base de como será implementado, independente da linguagem escolhida.

#### 6.1.3.3 - Projeto de Filtros IIR com método da Transformação Bilinear

O “Projeto de Filtros IIR com método da transformação bilinear” é chamado, no projeto, de BZT.

De acordo com a teoria da Transformação Bilinear Z, sabe-se que o Aluno deve entrar com as posições dos polos e zeros no plano S (de um filtro previamente calculado), pedir para o sistema transformar para o plano Z e em seguida visualizar o efeito daquele sistema sobre os sinais, em fase e amplitude. Neste caso, o ator Sistema é responsável por efetuar os cálculos e mostrar para o Aluno gráficos destes comportamentos.

Sendo assim, o diagrama de Caso de Uso do BZT se dá conforme a Figura 29.



*Figura 29 - Diagrama de Caso de Uso do BZT.*

No diagrama de Caso de Uso do BZT podemos observar as ações que são executadas pelo Aluno e pelo Sistema. Neste caso o Aluno entra com dados de um filtro pré-projetado no plano S, inserir os polos e os zeros no plano S e realizar a conversão. Por fim, o aluno pode requisitar a equação de diferenças para que possa utilizar em seus projetos o filtro criado. Nota que o Aluno não projeta este filtro, neste processo ele já possui os dados do filtro previamente projetado no plano S, mas nada impede que ele projete um filtro e utilize neste processo.

O Sistema tem o papel de calcular a transformação e mostrar para o Aluno o que acontece com os dados que o Aluno informou.

Por fim, com todos os casos do BZT observado, encontra-se no APÊNDICE I – DIAGRAMAS DE MODELAGEM os Diagramas de Atividade do BZT, Diagrama de Atividade da Atividade “Converte” e o Diagrama de Classe do sistema. As atividades simples como “Botão Reset”, “Retorna”, etc não são detalhados pela simplicidades da função. As classes que já foram detalhadas nas seções anteriores não são reescritas no projeto, pois como cada fase do projeto necessita da anterior, as classes são compartilhadas e não há a necessidade de refazer esta modelagem.

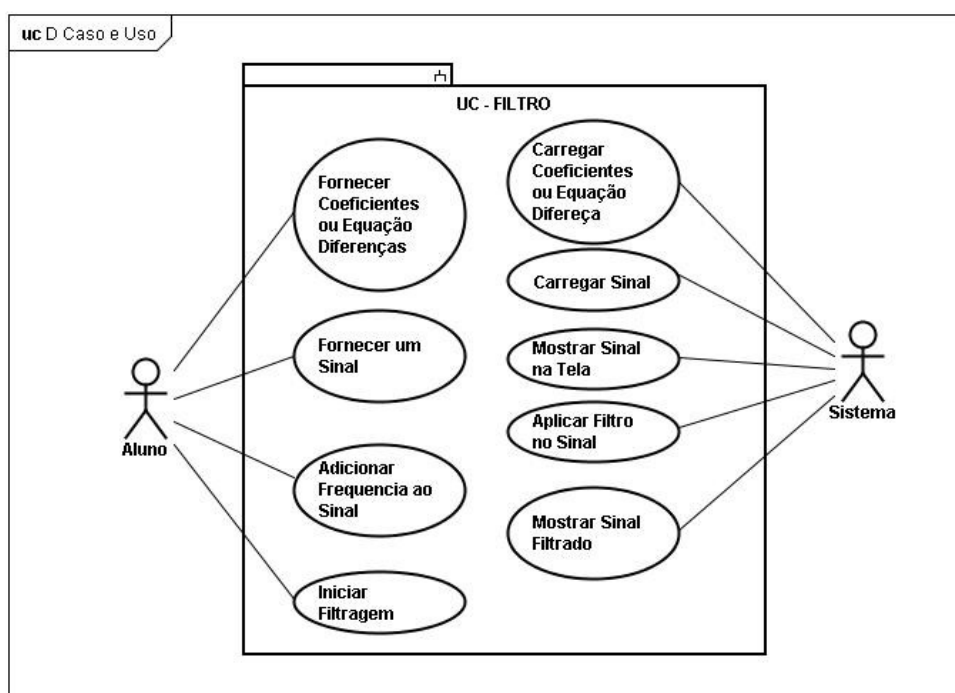
Com estas informações, tem-se conhecimento do que o sistema será capaz de realizar, quem irá atuar, como irá funcionar e a base de como será implementado, independentemente da linguagem escolhida.

#### 6.1.3.4 - Filtragem de Sinais utilizando coeficientes de filtros

O “Filtragem de Sinais utilizando coeficientes de filtros” é chamado, no projeto, de FILTRO.

O Aluno deve fornecer como dado de entrada os coeficientes ou a equação de diferença de um filtro, um sinal e pedir para o sistema calcular o sinal de saída. Neste caso o ator Sistema é responsável por efetuar os cálculos e mostrar gráficos para o Aluno.

Sendo assim, o diagrama de Caso de Uso do FILTRO se dá conforme a Figura 30.



*Figura 30 - Diagrama de Caso de Uso do FILTRO.*

No diagrama de Caso de Uso do FILTRO podemos observar as ações que são executadas pelo Aluno e pelo Sistema. Neste caso o Aluno deve fornecer os dados de um filtro projetado anteriormente, fornecer um sinal e pedir para o sistema calcular o sinal resultante. O Sistema tem o papel de calcular a transformação e mostrar para o Aluno o que acontece com os dados que o Aluno informou.

Por fim, com todos os casos do FILTRO observados, encontram-se no APÊNDICE I – DIAGRAMAS DE MODELAGEM o “Diagrama de Atividades do FILTRO”, “Diagrama de Atividade da Atividade Ler”, “Diagrama de Atividade da Atividade Cria Filtro”, “Diagrama de Atividade da Atividade Filtrar” e o “Diagrama de Classe do FILTRO”. As classes que já foram detalhadas nas seções anteriores não são reescritas no projeto, pois como cada fase do projeto necessita da anterior, as classes são compartilhadas e não há a necessidade de refazer esta modelagem.

Com estas informações, tem-se conhecimento do que o sistema será capaz de realizar, quem irá atuar, como irá funcionar e a base de como será implementado, independentemente da linguagem escolhida.

#### 6.1.4 - Implementação

A implementação é a fase em que os tutoriais foram escritos e as ferramentas programadas. Nos sub-tópicos são detalhadas cada fase do processo.

##### 6.1.4.1 - Implementação das Funções para o Scilab

O Scilab, por sua vez, não possibilita a implementação de classes de acordo com a UML, ou seja, não permite uma implementação com Orientação a Objetos. Devido a isso, para seguir a modelagem, foram separado em quatro grupos de funções que precisariam ser implementadas para atingir o objetivo.

Como estas funções são utilizadas no Mão na Massa do Portal Saúde+Educação<sup>TM</sup>, o conjunto de ferramentas pode ser implementado independentemente, cada função efetuando a sua tarefa, no fim, guiados pelos tutoriais, o aluno poderá utilizar adequadamente cada uma das ferramentas.

Cada conjunto de ferramentas leva o mesmo nome, conforme projeto visto na modelagem. A seguir apresenta-se uma descrição de cada conjunto de ferramentas desenvolvido no Scilab:

##### **PlanoZ:**

Plano Z é um conjunto de ferramentas que possibilita a visualização dos polos e zeros no plano Z. Este conjunto de funções recolhe um conjunto de pares de valores, onde o par representa o número em sua quantidade real e imaginária e com este conjunto separados em polos e zeros, calcula a atenuação em magnitude proporcional e decibéis, fase e apresenta todas estas informações

em um gráfico.

O cálculo da magnitude e da defasagem são feitos através do método geométrico, visto em Ifeachor e Jervis (1993). Depois, a função de plotagem calcula (a mesma que plota o plano Z) em decibéis para plotar na tela. As funções que fazem estes cálculos são chamadas separadamente da função que plota os gráficos.

As funções implementadas são as seguintes:

- eq\_diff: calcula a equação de diferenças de um sistema a partir dos polos e zeros.
- exp\_calc: calcula uma multiplicação polinomial.
- Filtra: aplica um sinal a uma equação de diferenças calculada pela eq\_diff.
- geo\_ang\_z: calcula o ângulo formado entre a reta formada por dois pontos e o eixo X do primeiro quadrante.
- geo\_dist\_z: calcula a distância geométrica entre dois pontos.
- geo\_modulo\_z: calcula a resposta em frequência, amplitude, de um sistema dados pelos polos e zero.
- geo\_teta\_z: calcula a resposta em frequência, fase, de um sistema dados pelos polos e zeros.
- Normalizateta: melhora a visualização de um gráfico dado pela função geo\_teta\_z.
- plota\_tela\_Z: exibe na tela o gráfico calculado pelas funções geo\_teta\_z e geo\_modulo\_z.

Para detalhes destas funções, foi criado um arquivo de ajuda para cada função seguindo o padrão do Scilab. Estes textos de ajuda podem se encontrados tanto no pacote fornecido no Portal Saúde+Educação<sup>TM</sup> quanto no APÊNDICE II – TEXTO DE AJUDA DAS FUNÇÕES DO SCILAB.

### **FiltroFIR:**

FiltroFIR é um conjunto de ferramentas que possibilita a criação de um filtro digital através do método de janelas. Este método consiste na criação matemática do filtro, e após o cálculo ele é transferido para o plano do tempo para poder ser aplicado ao sinal através da convolução. Acontece que nessa transição torna-se presente no filtro o fenômeno de Gibbs, então essa janela é criada para

amenizar este problema.

A ferramenta FiltroFIR permite que o usuário crie um filtro com os dados de entrada, por exemplo: tamanho da janela, tipo de janela, tipo de filtro, frequências de corte. A ferramenta também possibilita a visualização da resposta em frequência.

Nesta etapa, a visualização da fase não é possível, pois para se calcular a fase é necessário o valor dos polos e zeros, o que não é possível regredir possuindo apenas a resposta em frequência. Porém, como visto anteriormente, a fase não é um fator incômodo para os filtro FIR calculados por este método, pois estes filtros possuem fase linear.

As funções implementadas são as seguintes:

- convlution: função que realiza a operação convolução.
- cria\_filtro: função que calcula os coeficientes a partir dos dados de entrada.
- get\_att: função que retorna a atenuação de um filtro.
- janelamento: função que cria a janela. Utilizada pela função cria\_filtro.
- varredura: função que calcula a resposta em frequência do sistema.

Para detalhes destas funções, foi criado um arquivo de ajuda para cada função seguindo o padrão do Scilab. Estes textos de ajuda podem se encontrados tanto no pacote fornecido no Portal Saúde+Educação<sup>TM</sup> quanto no APÊNDICE II – TEXTO DE AJUDA DAS FUNÇÕES DO SCILAB.

### **BZT:**

BZT é um conjunto de ferramentas que possibilita a criação de um filtro digital através do método da transformação bilinear. Este método consiste na transformação de um filtro do plano S para o plano Z.

A Ferramenta BZT permite que o usuário crie um filtro dando como entrada os polos e zeros de um filtro no plano S.

As funções também permitem visualizar a posição dos polos e zeros no plano Z, resposta em amplitude e fase.

As funções implementadas são as seguintes:

- BZT: função que realiza a transformação bilinear Z.
- BZT\_ImagCalc: função que aplica a expressão da Transformação em uma equação de segundo grau.
- root2: função que calcula as raízes de uma função de segundo grau, cuja raiz resulta em um número imaginário (Delta Negativo). Em adição, o retorno resulta em valores separados para parte real e imaginária.

Para detalhes destas funções, foi criado um arquivo de ajuda para cada função seguindo o padrão do Scilab. Estes textos de ajuda podem se encontrados tanto no pacote fornecido no Portal Saúde+Educação<sup>TM</sup> quanto no APÊNDICE II – TEXTO DE AJUDA DAS FUNÇÕES DO SCILAB.

#### 6.1.4.2 - Implementação de Ferramentas de apoio para Aulas utilizando Scilab

Algumas ferramentas de apoio foram criadas para auxiliar os tutoriais. São as seguintes:

##### **Extract:**

Operações que preparam sinais para serem utilizados pelo Scilab. Ela abre sinais brutos, prepara em vetores e, em seguida, salva estes sinais em um formato de maneira que futuramente ele possa ser aberto sem a necessidade de processamento prévio para prepará-lo.

##### **Roteiros:**

Este conjunto de operações são rotinas executáveis (.SCE) que serão utilizados pelos alunos nas aulas práticas. Elas consistem em um conjunto de instruções com um espaço para o Aluno inserir os seus dados, ou seja, os que ele precisar para realizar a aula prática.

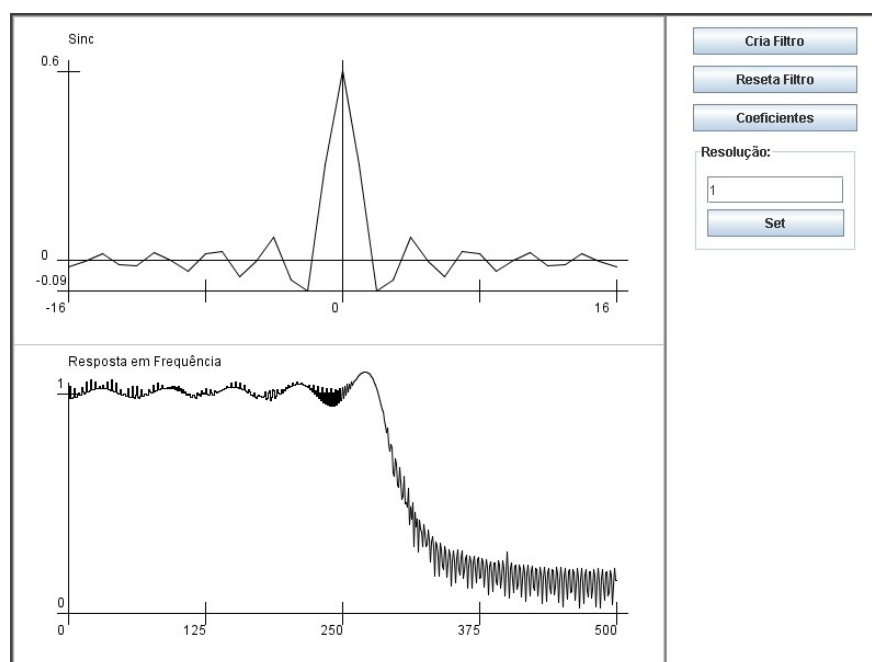
#### 6.1.4.3 - Implementação dos Java Applets

Para a implementação dos Java Applets foi necessário a utilização do Kit de Desenvolvimento Java (JDK) e um aplicativo de interface de desenvolvimento (IDE). Neste caso, optou-se pelo “*NetBeans IDE 6.1*”.

O NetBeans possui uma ferramenta que possibilita a execução e depuração de Applets sem a necessidade de uma interface *WEB* em funcionamento. Esta ferramenta facilita os testes e permite a depuração em tempo real, o que não seria possível se cada teste fosse efetuado abrindo um arquivo HTML. Os Applets foram implementados seguindo a modelagem e preparados para o ambiente *Web* e as telas do protótipo são detalhadas a seguir:

### **Telas do Applet o Filtro FIR pelo método de Janelas:**

O Applet para projeto de filtros FIR pelo método janelas possui uma janela principal, que é a tela que carrega quando o Applet se inicia. Essa janela é, graficamente, conforme apresentada na Figura 31.



*Figura 31 - Tela Principal do Applet para filtros FIR.*

Nesta tela, o gráfico do Sinc é apresentado na parte superior, conforme o título. Na parte inferior é apresentada a resposta em frequência do filtros projetados. Na barra lateral direita, onde estão os botões, encontram-se três botões e um letreiro “Resolução:”, com mais um botão internamente.

O botão Set, dentro do letreiro “Resolução:” tem o efeito de aplicar o valor que encontra-se no campo de texto. Esta é a resolução que define a qualidade dos gráficos apresentado à esquerda.

O botão Reseta Filtro, quando pressionado, limpa toda a tela e todas as configurações já feitas anteriormente.

Para se criar um filtro, é necessário pressionar o botão Cria Filtro, que quando pressionado apresenta a tela que se encontra na Figura 32.

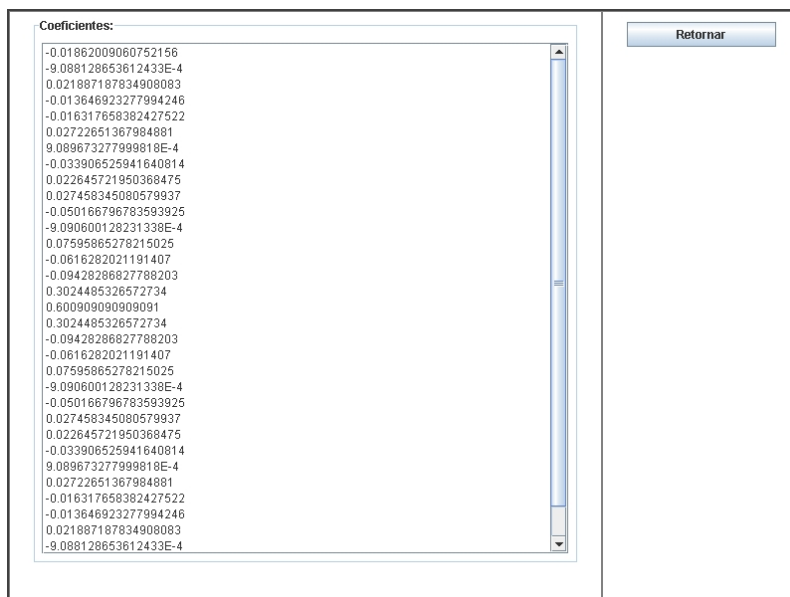


The screenshot shows a software interface for configuring an FIR filter. The top section contains several input fields and dropdown menus. On the left, there is a 'Filtro:' dropdown menu currently set to 'Passa Baixa'. Below it are two text input fields labeled 'Fc1:' and 'Fc2:', with values '0' and '250' respectively. In the center, there is a text input field labeled 'FS:' with the value '1000'. To the right of 'FS:', there is a 'Janela:' dropdown menu currently set to 'Retangular'. Below it is a text input field labeled 'N(Tap):' with the value '33'. On the far right, there are two buttons: 'Aceitar' (Accept) and 'Cancelar' (Cancel). Below the top section is a large gray rectangular area, which likely displays the filter's frequency response or coefficients.

*Figura 32 - Tela de configuração dos Filtros FIR.*

Neste tela, podemos entrar com os dados e pressionar em Aceitar. Se os dados estiverem corretos, a tela principal, Figura 31, é apresentada novamente com os dados do filtro. Se os dados estiverem errados ou o botão Cancelar for pressionado, a tela principal é apresentada sem os resultados.

Na tela principal, Figura 31, quando o botão Coeficientes for pressionado é mostrada a tela da Figura 33.

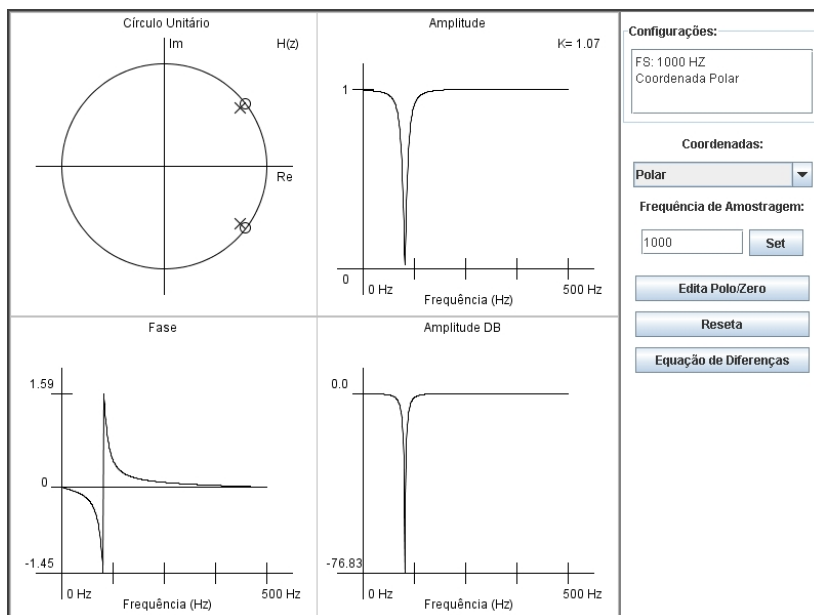


*Figura 33 - Tela de Coeficientes do filtro FIR.*

Nesta tela, se o filtro já estiver configurado, os coeficientes serão apresentados. Se o filtro não estiver projetado, a tela é mostrada mas a lista estará vazia. Esta lista foi desenvolvida de uma maneira que é possível selecionar todo o conteúdo e copiar para a *Área de Transferência* do sistema operacional.

#### **Telas do Applet o Filtro IIR pelo método de inserção de Polos e Zeros:**

A tela principal e inicial do Applet dos Filtros IIR pelo método de inserção é apresentada na Figura 34.



*Figura 34 - Tela Principal do PlanoZ.*

Esta tela apresenta os quatro gráficos úteis para o resultado de um filtro projetado por este método. Quando iniciado sem o projeto do filtro, apenas o círculo unitário (Superior esquerda) é visualizável, porém, sem os polos e zeros.

Na seção da esquerda, onde há os gráficos, é possível visualizar o círculo unitário do Plano Z (Superior esquerda), a magnitude (Superior direita), a defasagem causada pelo sistema (Inferior esquerda) e a magnitude em decibéis (Inferior direita).

Na área de botões, à direita, podemos visualizar algumas configurações importantes dos sistemas (no quadro “Configurações:”), como frequência de amostragem (FR:) e as coordenadas que os polos e zeros foram inseridos. Na área “Frequência de amostragem:” é possível mudar a frequência de amostragem do sistema, inserindo o valor e pressionando o botão Set. O botão Reseta limpa todas as configurações do sistema. O botão Edita Polo/Zero leva o usuário para a tela de configuração do filtro, que é mostrada na Figura 35. O botão Equação de Diferenças leva o usuário para a tela que mostra a equação de diferenças para o usuário, conforme a Figura 36.

Coordenadas: Polar 0.937

Polo/Zero: Polo 37

Módulo

Ângulo

Inserir Polo/Zero

Remove Polo/Zero

Limpa Tabela

Aceitar

Cancelar

Polos:

Módulo	Ângulo
--------	--------

Zeros:

Módulo	Ângulo
1.0	37.0
1.0	-37.0

Figura 35 - Tela de configuração do PlanoZ.

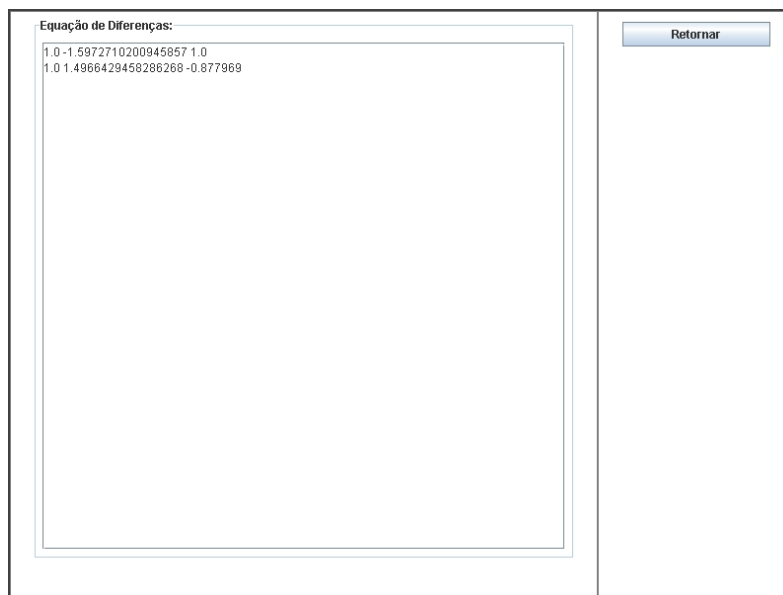
Na Figura 35, o usuário pode adicionar polos e zeros do sistema.

O usuário pode escolher o que ele deseja adicionar, selecionando no *combo-box* Polo/Zero. Quando um valor for adicionado neste *combo-box* e houver o valor “zero”, o adicionado será um zero, consequentemente aparecerá na tabela de **zeros**. Se for “polo”, será adicionado um polo e o valor aparecerá na tabela de **polos**.

No *combo-box* Coordenadas é possível escolher de qual maneira o usuário deseja entrar com os valores, se é em coordenadas polares ou retangulares. Quando selecionado Polar, os valores de entrada são módulo e ângulo. Quando for selecionado retangular, os valores de entrada são valor real e valor imaginário.

Quando o valor é digitado nos dois campos, o botão Inserir Polo/Zero se disponibiliza e quando pressionado a rotina de inserção é executada. Quando um polo ou zero da tabela é pressionado, o botão Remover Polo/Zero se disponibiliza e quando for pressionado a rotina de remoção de polo ou zero é executada. O botão Limpa Tabela elimina todos os polos e zeros do sistema.

No fim, quando o usuário tiver inserido seus polos e zeros, se ele clicar em Aceitar, o Applet retorna para a tela principal, executando o cálculo. Caso o usuário pressione Cancelar, todas as mudanças serão ignoradas e o Applet retornará para a tela principal sem modificações.



*Figura 36 - Tela dos coeficientes do PlanoZ.*

A tela de Equação de Diferenças se assemelha muito a tela de coeficientes (Figura 33). A diferença é apenas que a equação de diferenças é transposta antes de aparecer na tela. Então tem-se duas linhas e o conteúdo dessas linhas distribuído em colunas. Eles também podem ser copiados para a *Área de Transferência* do sistema operacional.

#### **Telas do Applet o Filtro IIR pelo método da Transformação Bilinear Z:**

A tela principal do Applet da BZT é apresentada na Figura 37.

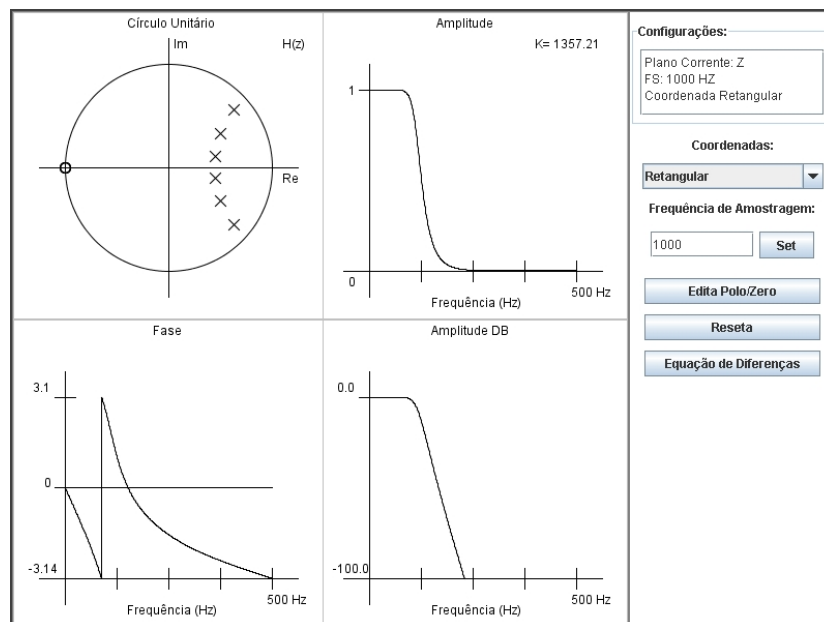


Figura 37 - Tela principal da BZT.

A tela deste Applet é semelhante a tela do Applet PlanoZ. Uma das diferenças é que é possível visualizar o plano corrente no painel “Configurações:”, na lateral direita, que informa ao leitor se os dados já foram convertidos para o plano Z.

A tela de configurações é apresentada quando pressionado o botão Edita Polo/Zero, conforme figura 38.

Figure 38 shows the configuration screen of the BZT applet. It includes fields for 'Coordenadas' (Polar) and 'Polo/Zero' (Polo). Below these are tables for 'Polos' and 'Zeros' with columns for 'Módulo' and 'Ângulo'. Buttons for 'Inserir Polo/Zero', 'Remove Polo/Zero', 'Limpa Tabela', 'Converte Para Z', 'Aceitar', and 'Cancelar' are also visible.

Polos:		Zeros:	
Módulo	Ângulo	Módulo	Ângulo
0.766	15.0	1.0E100	90.0
0.766	-15.0	1.0E100	-90.0
0.766	45.0	1.0E100	90.0
0.766	-45.0	1.0E100	-90.0
0.766	75.0	1.0E100	90.0
0.766	-75.0	1.0E100	-90.0
0.766	105.0	1.0E100	90.0
0.766	-105.0	1.0E100	-90.0
0.766	135.0	1.0E100	90.0
0.766	-135.0	1.0E100	-90.0
0.766	165.0	1.0E100	90.0
0.766	-165.0	1.0E100	-90.0

Figura 38 - Tela de configuração da BZT, antes da conversão.

A tela de configuração do BZT também é muito semelhante à tela de configuração do PlanoZ. Porém, neste Applet os polos e zeros são adicionados no plano S. O que faz surgir o botão Converter Plano Z. Este botão, quando pressionado transforma todos os polos e zeros do sistema para o plano Z. Na Figura 39, pode ser visualizada a mesma tela após a conversão. Quando a conversão é realizada, o botão de conversão se apaga e não fica mais disponível até a próxima vez que o Applet for carregado.

Coordenadas:  Parte Real

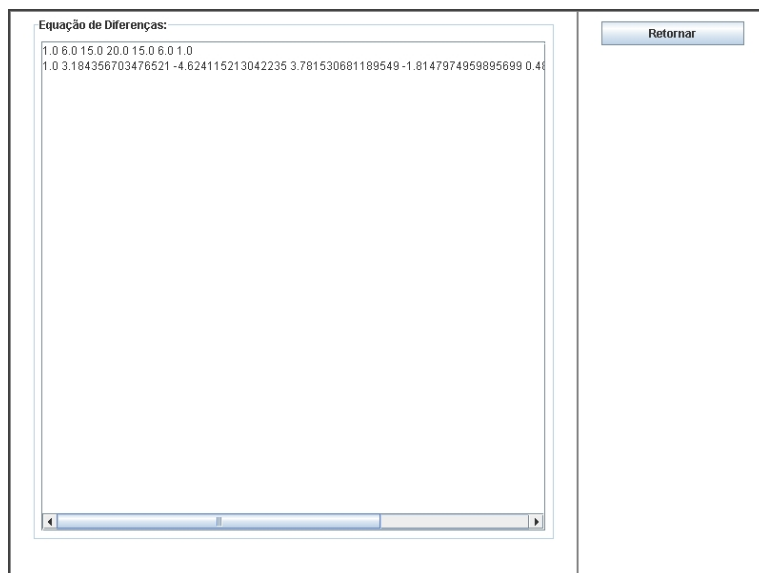
Polo/Zero:  Parte Imaginária

Polos:	
Parte Real	Parte Imaginária
0.6344582030792338	0.5501336629508737
0.6344582030792338	-0.55013366295087...
0.5054163508734439	0.32081577529562...
0.5054163508734439	-0.32081577529562...
0.45230379778558...	0.10508673294022...
0.45230379778558...	-0.10508673294022...

Zeros:	
Parte Real	Parte Imaginária
-1.0	-0.0
-1.0	-0.0
-1.0	-0.0
-1.0	-0.0
-1.0	-0.0
-1.0	-0.0

*Figura 39 - Tela de configurações do BZT, depois da conversão.*

Depois de convertido, com o botão Aceitar pode-se retornar à tela principal e visualizar os resultados. Se for pressionado Cancelar ou Aceitar sem realizar a conversão, nada é visualizado, pois os polos e zeros do plano Z ainda não foram calculados.

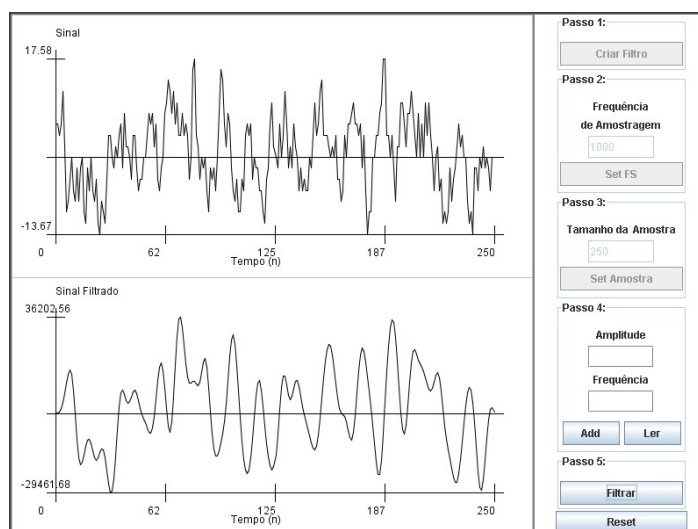


*Figura 40 - Tela da Equação de Diferenças do BZT.*

A tela de equação de diferenças do BZT é idêntica a do PlanoZ, uma vez que após a conversão, os dados dos dois sistemas são iguais.

### **Telas do Applet de Filtragem de Sinais:**

A tela principal de filtragem é apresentada na Figura 41.



*Figura 41 - Tela principal com filtro criado e sinal carregado e filtrado.*



Nesta tela, pode-se notar alguns botões apagados e outros ativos. A sequência de ações é passo-a-passo, onde o “Passo 1” é executado primeiro e nos outros passos os botões ficam desativados. Quando o “Passo 1” termina, ele se desativa e ativa o próximo. Estes passos podem ser vistos na lateral direita. Cada área contornada e intitulado por “Passo X” faz parte daquele passo.

Quando o Criar Filtro, no “Passo 1” é pressionado a tela da Figura 42 apresentada.

A interface da tela de criação de filtro é organizada em uma grade. No topo, há três painéis de progresso:

- Passo 1:** Contém o texto "Inserir Valores dos coeficientes do filtro na região a baixo." e um botão "Carregar" desativado.
- Passo 2:** Contém o texto "Quando estiver pronto, clique no botão 'Carregar' para carregar o filtro." e um botão "Carregar" desativado.
- Passo 3:** Contém o texto "Carregamento:" e "Sucesso!" em um fundo verde, com um botão "Carregar" desativado.

À direita dos painéis, há dois botões: "Aceitar" (ativo) e "Cancelar" (desativado). Abaixo dos painéis, há uma área de entrada de dados com o seguinte texto:

```
1.0 6.0 15.0 20.0 15.0 6.0 1.0
1.0 3.184356703476521 -4.624115213042235 3.781530681189549 -1.8147974959895699
0.4803608112413413 -0.05449107050696304
```

*Figura 42 - Tela de criação de filtro.*

Na tela de criação de filtros, os coeficientes e equações de diferenças dos filtros projetados pelos outros Applets (Ou criado manualmente se estiver no padrão) podem ser inseridos e carregados. Quando os valores são escritos na quadro inferior, o botão Carregar se disponibiliza. Quando pressionado, o sistema avalia o valor e informa com sucesso ou falha. Caso haja sucesso, o botão Aceitar se torna disponível. Caso haja falha, o botão Aceitar não se disponibiliza e o “Passo 1” se ativa novamente.

Quando pressionado o botão Aceitar, o sistema volta para a tela principal e ativa o “Passo 2”.

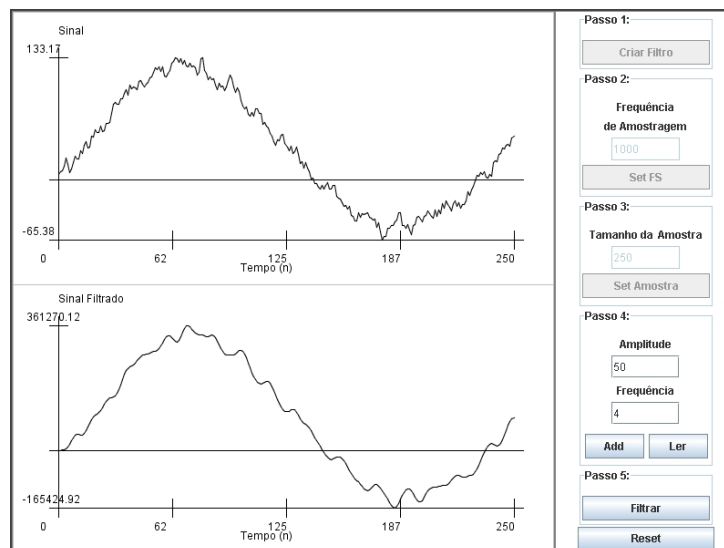
Voltando para a tela principal, o “Passo 2” e o “Passo 3” são executados sem a necessidade de uma outra tela. No “Passo 4”, se pressionando-se o botão Ler, a tela da Figura 43 é apresentada.

<b>Passo 1:</b> Inserir Sinal na região a baixo.	<b>Passo 2:</b> Escolha no ComboBox o canal que deseja carregar. Canal 5.0	<b>Passo 3:</b> Carregamento: <b>Sucesso!</b>	Aceitar Cancelar
<pre> -1.0000 -2.0000 -3.0000 -4.0000 -5.0000 -6.0000 -7.0000 -8.0000 -9.0000 -10.0000 -11.0000 -12.0000 -13.0000 -14.0000 -15.0000 -16.0000 -17.0000 -18.0000 -19.0000 -20.0000 -21.0000 -22.0000 -23.0000 -24.0000 -25.0000 -26.0000 -27.0000 -28.0000 -29.0000 -30.0000 -31.0000 -32.0000 29.2968 5.8594 -19.5312 -27.3437 5.8594 -9.7656 15.6250 -21.4844 35.1562 7.8125 3.9062 -46.8750 0.0000 3.9062 3.9062 -1.9531 42.9687 -9.7656 -13.6719 -25.3906 1.9531 -7.8125 5.8594 5.8594 23.4375 5.8594 0.0000 -33.2031 3.9062 1.9531 7.8125 -1.9531 33.2031 7.8125 -29.2968 -21.4844 5.8594 -13.6719 13.6719 -23.4375 33.2031 7.8125 -1.9531 -42.9687 1.9531 7.8125 3.9062 0.0000 41.0156 -3.9062 -15.6250 -27.3437 1.9531 -5.8594 7.8125 5.8594 29.2968 0.0000 0.0000 -31.2500 0.0000 1.9531 3.9062 3.9062 39.0625 15.6250 -15.6250 -46.8750 3.9062 -7.8125 11.7187 -27.3437 41.0156 13.6719 5.8594 -52.7343 -3.9062 0.0000 -1.9531 9.7656 39.0625 0.0000 -5.8594 -17.5781 -1.9531 -9.7656 11.7187 3.9062 41.0156 7.8125 -3.9062 -37.1093 -5.8594 -1.9531 0.0000 13.6719 37.1093 19.5312 -15.6250 -58.5937 5.8594 -3.9062 11.7187 -31.2500 42.9687 11.7187 5.8594 -44.9218 1.9531 0.0000 1.9531 11.7187 41.0156 -9.7656 -5.8594 -11.7187 7.8125 -9.7656 9.7656 44.9218 7.8125 0.0000 -37.1093 1.9531 -5.8594 9.7656 15.6250 31.2500 27.3437 -31.2500 -44.9218 11.7187 -11.7187 17.5781 -41.0156 44.9218 19.5312 13.6719 -35.1562 -3.9062 5.8594 7.8125 9.7656 37.1093 -1.9531 1.9531 1.9531 -1.9531 -7.8125 17.5781 11.7187 41.0156 13.6719 9.7656 -21.4844 -7.8125 5.8594 3.9062 9.7656 41.0156 21.4844 -33.2031 -33.2031 1.9531 -21.4844 17.5781 -29.2968 39.0625 17.5781 15.6250 -39.0625 5.8594 7.8125 5.8594 7.8125 39.0625 0.0000 1.9531 1.9531 7.8125 -11.7187 17.5781 11.7187 37.1093 5.8594 11.7187 -19.5312 7.8125 13.6719 0.0000 3.9062 48.8281 23.4375 -31.2500 -25.3906 -9.7656 -15.6250 13.6719 -21.4844 46.8750 15.6250 3.9062 -29.2968 -1.9531 0.0000 -1.9531 1.9531 31.2500 3.9062 -1.9531 -5.8594 -1.9531 -15.6250 9.7656 7.8125 46.8750 0.0000 -3.9062 -11.7187 0.0000 0.0000 -1.9531 1.9531 54.6875 19.5312 -25.3906 -41.0156 -7.8125 -17.5781 19.5312 -31.2500 44.9218 13.6719 3.9062 -41.0156 -3.9062 -3.9062 -1.9531 37.1093 1.9531 -5.8594 -11.7187 -3.9062 -23.4375 9.7656 3.9062 44.9218 0.0000 -3.9062 -17.5781 -3.9062 3.9062 -13.6719 3.9062 </pre>			

Figura 43 - Tela de carregamento de sinal no Applet.

Nesta tela, é possível carregar um sinal. Este sinal pode ser escrito no painel inferior. Quando inserido, o sistema automaticamente verifica se o formato está aceitável. Caso sim, o “Passo 2” se ativa e possibilita escolher o canal. Quando o canal é escolhido, o sistema carrega o número de amostras conforme informado na tela principal e carrega na memória. Depois disto, o Applet disponibiliza a opção de aceitar e voltar para a tela principal.

Além de carregar um sinal, é possível adicionar uma frequência qualquer. Para isto apenas é necessário entrar com os valores no “Passo 4” e pressionar o botão Add. Esse evento pode ser visualizado na Figura 44.



*Figura 44 - Adição de frequências senoidais ao sinal original.*

Após um sinal ser lido ou uma frequência ser adicionada, o “Passo 5” é ativado, possibilitando a filtragem. Este resultado aparece no painel inferior, conforme já está demonstrado nas Figuras 41 e 44.

## 6.2 - Escrita dos Conteúdos

Possuindo todas as ferramentas implementadas, os tutoriais foram escritos seguindo a fundamentação teórica com uma estrutura do conhecimento progressiva (dos conteúdos básicos para os mais complexos).

Estes tutoriais consistem na seguintes seções encontrada no Portal Saúde+Educação<sup>TM</sup>: Conteúdo, História, Correlação Clínica, Mão-na-Massa e Avaliação.

Cada seção desta (Conteúdo, História, Correlação Clínica, Mão-na-Massa e Avaliação) foram elaboradas com um conteúdo escrito baseado nesta fundamentação teórica e pode ser encontrado no site do Portal Saúde+Educação<sup>TM</sup> ([www.saudeeducacao.ufsc.br](http://www.saudeeducacao.ufsc.br)).

## 6.3 - Validação

A validação é o processo que vai garantir o funcionamento das ferramentas implementadas. Como se tratam de funções para filtragem digital, uma opção de garantir o funcionamento dos filtros é utilizar a transformada de Fourier. Com a transformada rápida de Fourier discreta, função

nativa do Scilab, podemos realizar uma transformada antes de se filtrar um sinal e uma logo após a filtragem, observando os resultados podemos saber exatamente qual o efeito do filtro sobre o sinal.

Como as ferramentas matemáticas foram inicialmente implementadas em Scilab e depois apenas reproduzidas em JAVA para os Applets, os testes serão executados apenas com as funções em Scilab. Isto porque se garante que em JAVA ou em Scilab as funções tem o mesmo comportamento.

Os testes se dividem em dois grupos, filtros FIR e filtros IIR.

### 6.3.1 - Validação das ferramentas para filtros FIR

A função que gera os coeficientes do filtro FIR é apenas uma função que os gera para todos os tipos de filtro FIR, dentre os apresentados na modelagem, identificando cada tipo por uma variável inteira na entrada desta função. Como dentro da função, conforme este valor fornecido na entrada, pode ser gerado um tipo diferente de filtro, há a necessidade de validar cada parte desta função.

Para mais detalhes da função, ela é capaz de gerar um filtro FIR dos seguintes tipos: passa alta, passa baixa, passa banda e rejeita banda. Estes quatro tipo de filtro que serão testados.

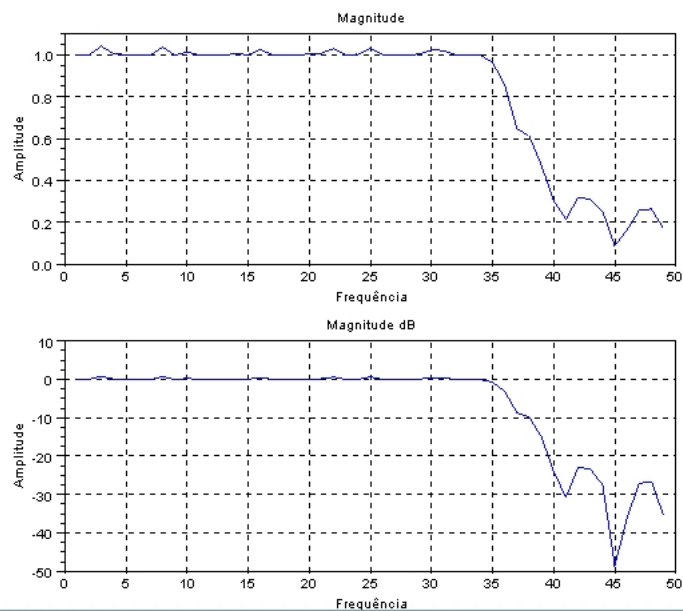
#### 6.3.1.1 - Passa Baixa

Os testes com o filtro passa baixa foram efetuados com as seguintes configurações:

Frequência de corte: 35 Hz

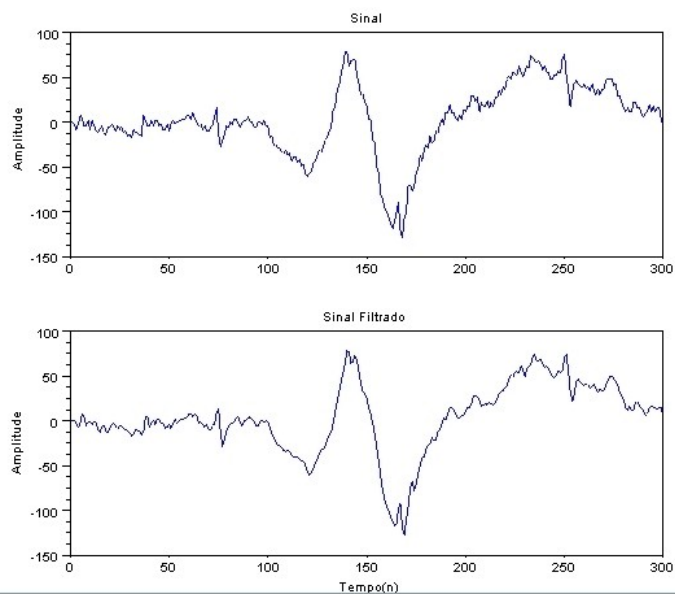
Frequência de amostragem: 100 Hz

A Figura 45 apresenta a resposta em frequência do filtro gerado pelo função.



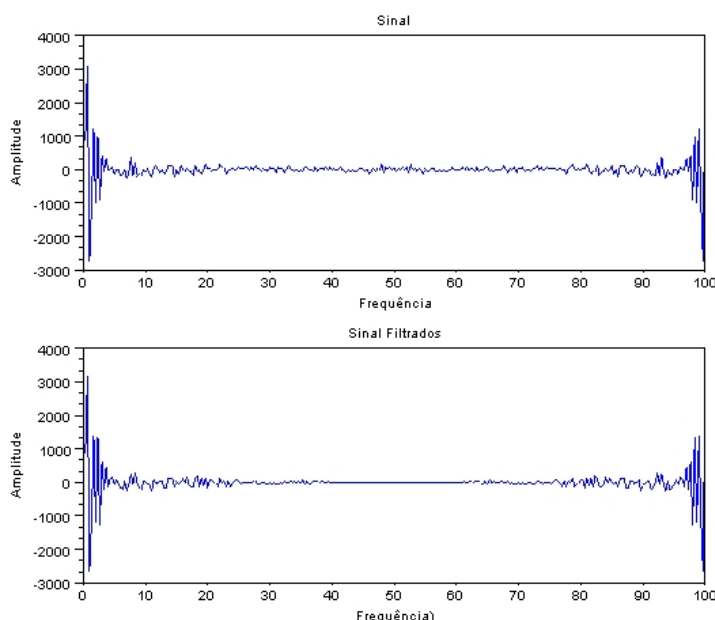
*Figura 45 - Resposta em Frequência do filtro passa baixa gerado com a função de gerar coeficientes para filtros FIR. Parte superior a resposta proporcional, na parte inferior a resposta em Decibéis.*

A Figura 46 apresenta o sinal sendo aplicado pelo filtro.



*Figura 46 - Sinal do EEG sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.*

A Figura 47 apresenta a Transformada de Fourier dos sinais.



*Figura 47 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.*

Na Figura 47 pode-se observar que o sinal original apresenta componentes em diversas frequências. O intervalo que apresenta a informação útil se encontra entre 0 e 50 Hz. Entre 50 e 100 Hz é o intervalo espelhado de 0 a 50 Hz. Este evento ocorre devido ao teorema de Nyquist. Frequências acima de 50 Hz não são amostradas e são removidas por um filtro anti-*aliasing*, que é um filtro analógico inserido antes da amostragem.

Observando os dois gráficos da Figura 47, nota-se que no sinal filtrado as frequências acima de 40 Hz desapareceram completamente. O filtro havia sido projetado para filtrar acima de 35 Hz. O que ocorre é que a banda de transição se inicia um pouco antes de 35 Hz. Na Figura 47, é possível visualizar que as frequências a partir de 35 Hz começam a se atenuar até chegar a um valor próximo de zero e assim seguindo até 50 Hz. Esta figura prova que o filtro está funcionando conforme o desejado.

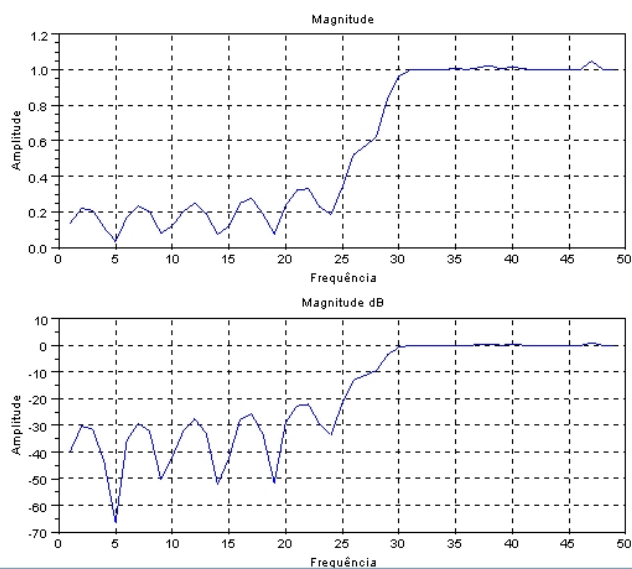
#### 6.3.1.2 - Passa Alta

Os testes com o filtro passa alta foram efetuados com as seguintes configurações:

Frequência de corte: 25 Hz

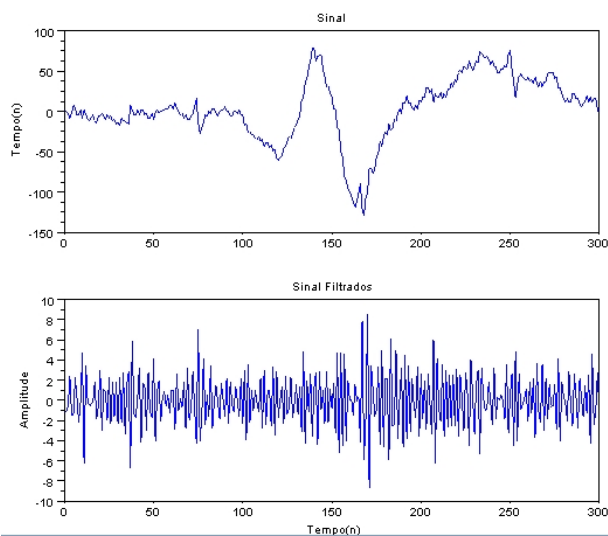
Frequência de amostragem: 100 Hz

A Figura 48 apresenta a resposta em frequência do filtro gerado pelo função.



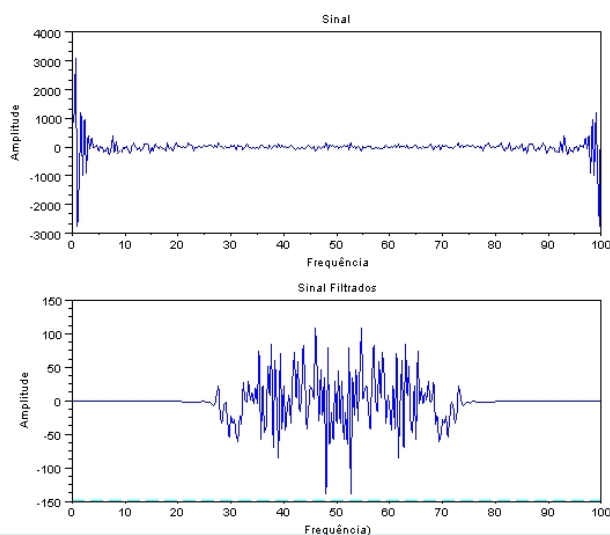
*Figura 48 - Resposta em Frequência do filtro passa alta gerado com a função de gerar coeficientes para filtros FIR. Parte superior a resposta proporcional, na parte inferior a resposta em Decibéis.*

A Figura 49 apresenta o sinal sendo aplicado pelo filtro.



*Figura 49 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo, o sinal filtrado.*

A Figura 50 apresenta a transformada de Fourier dos sinais.



*Figura 50 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e abaixo, a FFT após a aplicação do filtro.*

Observando os gráficos da Figura 50, podemos notar que as frequências abaixo de 25 Hz foram completamente eliminadas do sinal após a filtragem, o que comprova o funcionamento do filtro passa alta.

#### 6.3.1.3 - Passa Banda

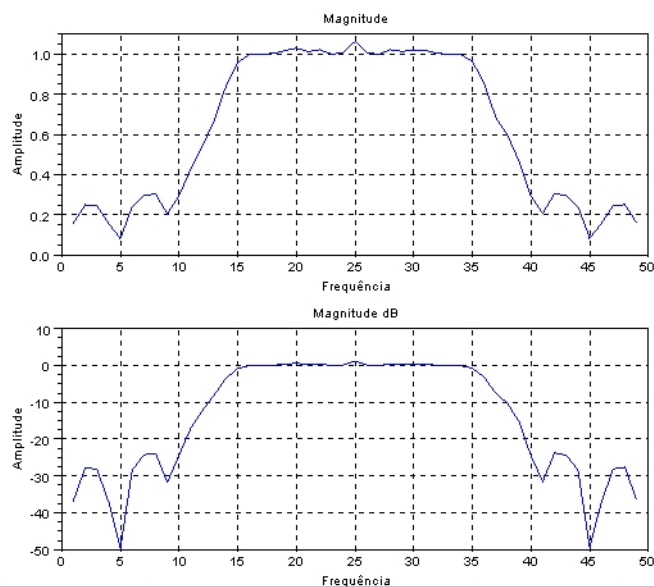
Os testes com o filtro passa banda foram efetuados com as seguintes configurações:

Frequência de corte: 15 Hz até 35 Hz

Frequência de amostragem: 100 Hz

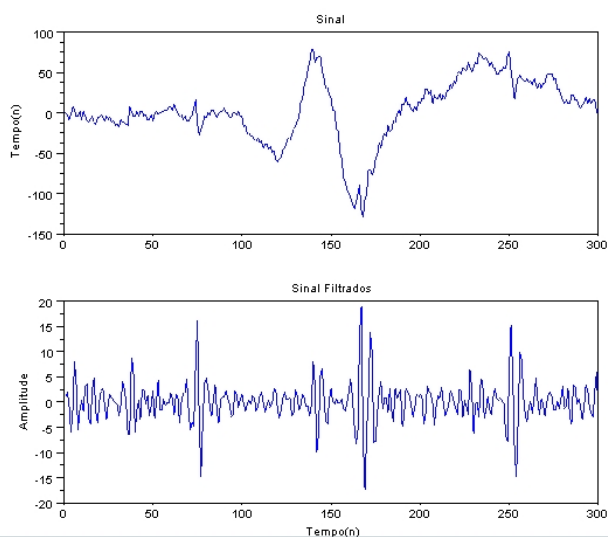
A Figura 51 apresenta a resposta em frequência do filtro gerado pelo função.





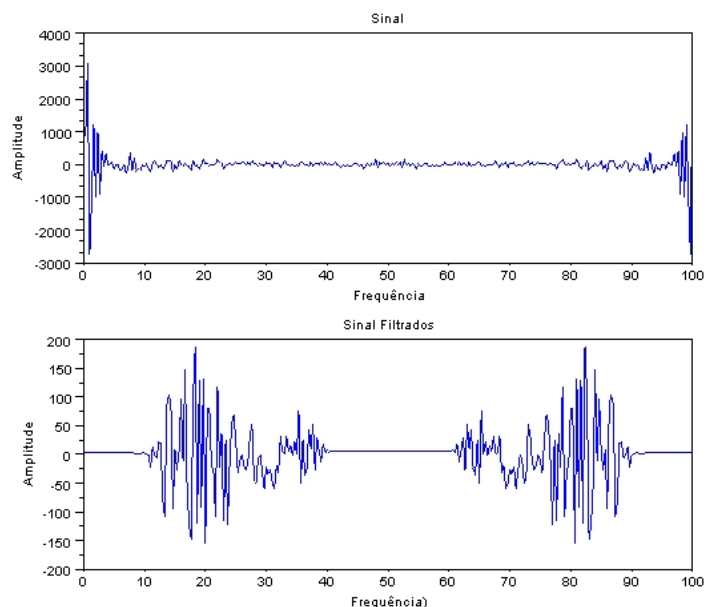
*Figura 51 - Resposta em Frequência do filtro passa Banda gerado com a função de gerar coeficientes para filtros FIR. Na parte superior, a resposta proporcional, na parte inferior, a resposta em Decibéis.*

A Figura 52 apresenta o sinal sendo aplicado pelo filtro.



*Figura 52 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.*

A Figura 53 apresenta a transformada de Fourier dos sinais.



*Figura 53 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.*

Observando os gráficos da Figura 53, podemos notar que as frequências abaixo de 15 Hz e acima de 35 Hz foram, de maneira gradativa, sendo eliminadas do sinal, até atingir um ponto em que as frequências (acima e abaixo) foram completamente eliminadas. Este fato comprova o funcionamento do filtro passa banda.

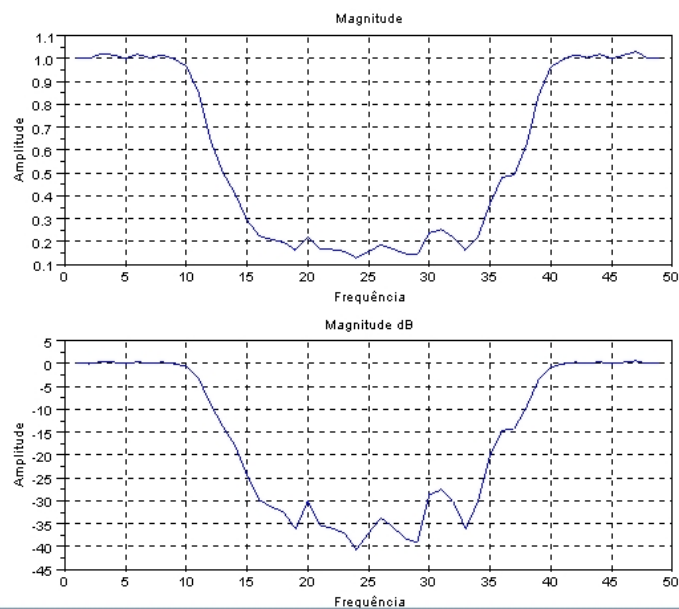
#### 6.3.1.4 - Rejeita Banda

Os testes com o filtro rejeita banda foram efetuados com as seguintes configurações:

Frequência de corte: 10 Hz até 40 Hz

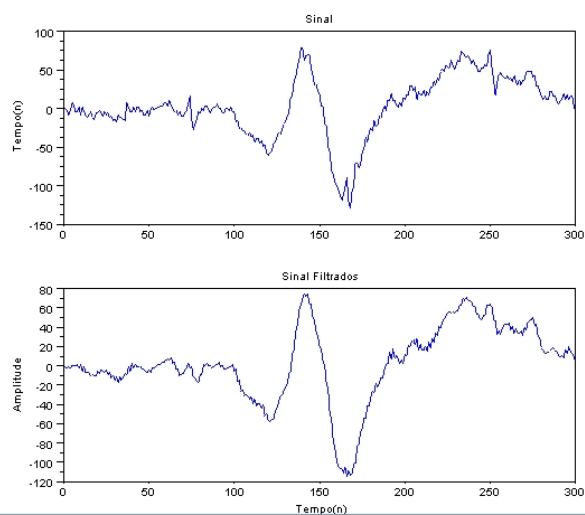
Frequência de amostragem: 100 Hz

A Figura 54 apresenta a resposta em frequência do filtro gerado pelo função.



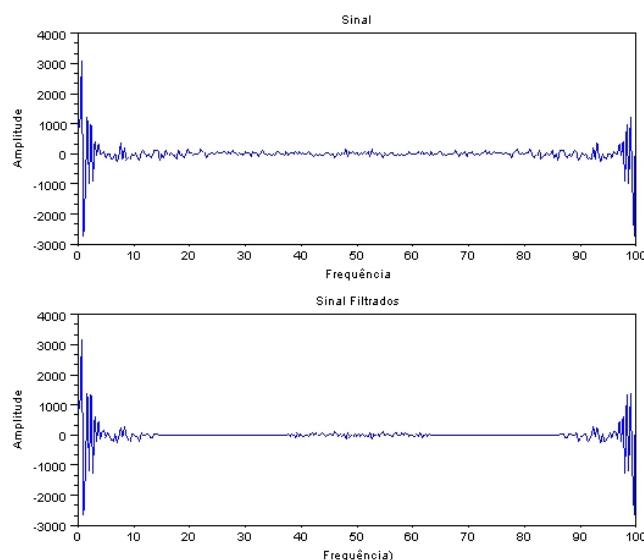
*Figura 54 - Resposta em Frequência do filtro rejeita banda gerado com a função de gerar coeficientes para filtros FIR. Na parte superior, a resposta proporcional, na parte inferior, a resposta em Decibéis.*

A Figura 55 apresenta o sinal sendo aplicado pelo filtro.



*Figura 55 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.*

A Figura 56 apresenta a Transformada de Fourier dos sinais.



*Figura 56 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e abaixo a FFT após a aplicação do filtro.*

Observando os gráficos da Figura 56, podemos notar que as frequências acima de 1 Hz e abaixo de 40 Hz foram, de maneira gradativa, sendo eliminadas do sinal, até atingir um ponto em que as frequências (entre) foram completamente eliminadas. Este fato comprova o funcionamento do filtro rejeita banda.

### 6.3.2 - Validação das ferramentas para filtros IIR

A Validação dos Filtros IIR se divide em duas etapas. A primeira etapa valida as funções de inserção de polos e zeros no plano Z. A segunda parte valida a transformação efetuada pela função de transformação bilinear.

#### 6.3.2.1 - Inserção de polos e zeros

A inserção de polos e zeros foi testada com a inserção de polos e zeros de um filtro Butterworth passa alta. Estes valores foram calculados e inseridos no *Shell* do Scilab conforme Figura 57.

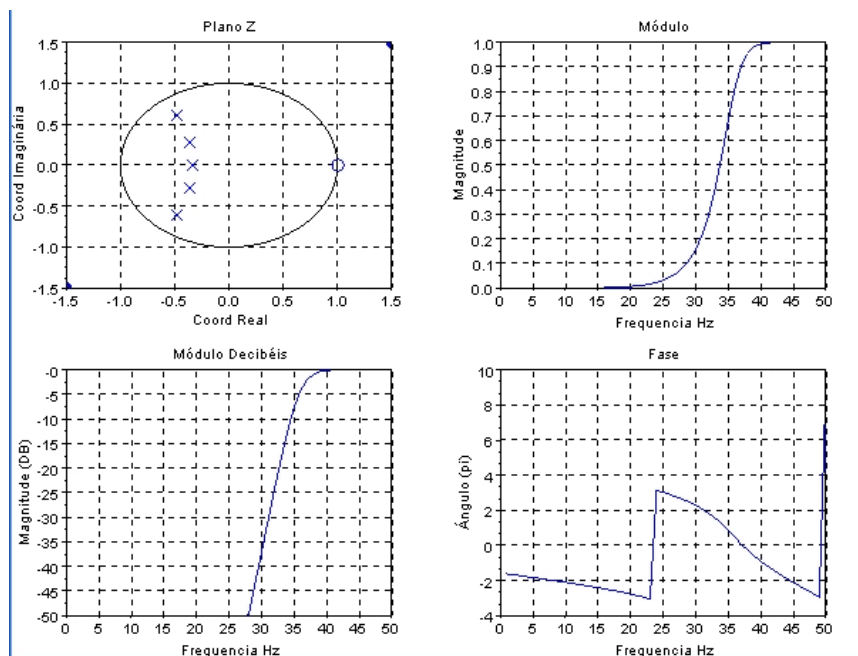
```

pz =
- 0.4782092    0.6115185
- 0.4782092   - 0.6115185
- 0.3618560    0.2859899
- 0.3618560   - 0.2859899
- 0.3311037    3.725D-09
zz =
1.    0.
1.    0.
1.    0.
1.    0.
1.    0.

```

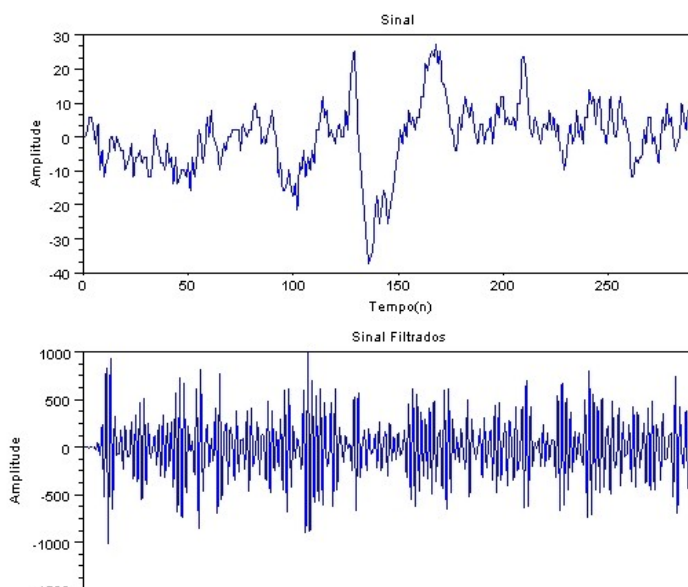
*Figura 57 - Polos e Zeros de um filtro Butterworth Passa Alta, no plano Z.*

Aplicando estes polos e zeros na função que calcula os coeficientes deste filtro, resultou em um filtro com a configuração conforme mostra a Figura 58.



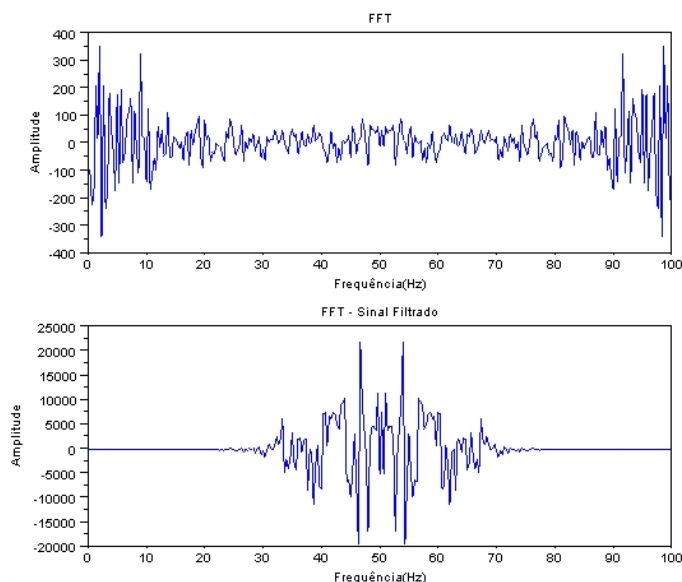
*Figura 58 - Configurações do filtro em questão.*

Um sinal foi escolhido para aplicação do sinal, para que posteriormente a FFT possa ser aplicada. O sinal antes e depois de ser filtrado pode ser visualizado na Figura 59.



*Figura 59 - Sinal sendo filtrado pelo filtro em questão. Acima o sinal natural e abaixo o sinal filtrado.*

E aplicando a FFT em cada sinal tem-se o que mostra a Figura 60.



*Figura 60 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. Acima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.*

Visualizando a Figura 58, conclui-se que o filtro em questão se trata de um Butterworth, passa alta com frequência de corte em aproximadamente 35 Hz. Com isso, observando a Figura 60, comprova-se que o filtro elimina as frequências menores que 35 Hz, gradativamente, o que comprova o funcionamento e a integridade dos coeficientes gerados.

### 6.3.2.2 - Transformação bilinear com Butterworth

As funções da transformação bilinear são testadas de maneira semelhante ao método de inserção de polos e zeros. A função de transformação é aplicada e logo em seguida, os polos e zeros, que são resultados da transformação, são inseridos na função e calculados. Uma vez que o modo de inserção de polos e zeros deve estar funcionando, a validação deste método não só o valida como também reforça o método de inserção.

A diferença básica entre este método e o de inserção é que neste os dados de entrada são os polos e os zeros no plano S.

Neste exemplo, como dado de entrada são utilizados os polos e zeros de um filtro Butterworth passa baixa. Estes valores foram calculados e inseridos no *Shell* do Scilab, conforme a Figura 61.

```

94
95 p = [-1.2298 +3.7852i;
96 -3.2198 +2.3393i;
97 -3.98 +0i]
98
99 z = [0 1D+100;
100 0 1D+100;
101 0 1D+100;
102 0 1D+100;
103 0 1D+100]

```

*Figura 61 - Polos e Zeros de um filtro Butterworth Passa Baixa, no plano S.*

Aplicando estes polos e zeros na função de transformação e em seguida nas funções que calculam os coeficientes deste filtro, resultou em um filtro com a configuração conforme mostra Figura 62.

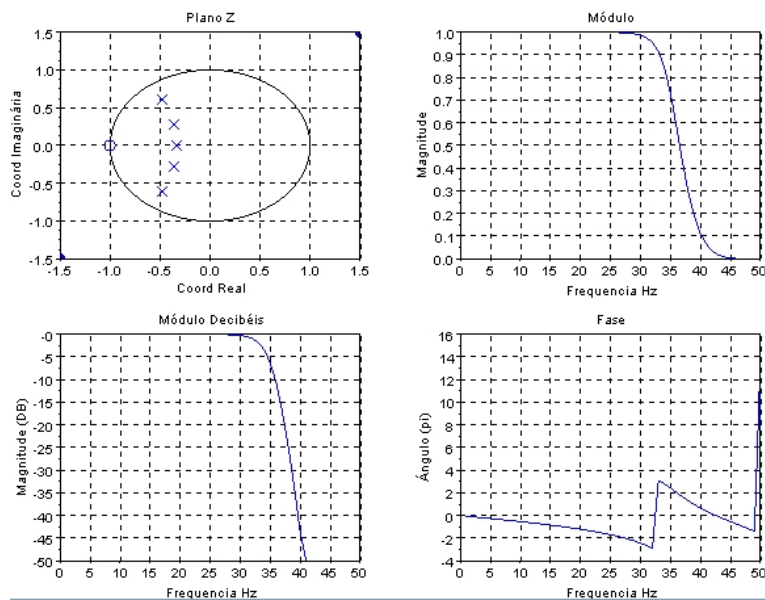


Figura 62 - Configurações do filtro em questão.

Um sinal foi escolhido para aplicação do sinal, para que posteriormente a FFT possa ser aplicada. O sinal antes e depois de ser filtrado pode ser visualizado na Figura 63.

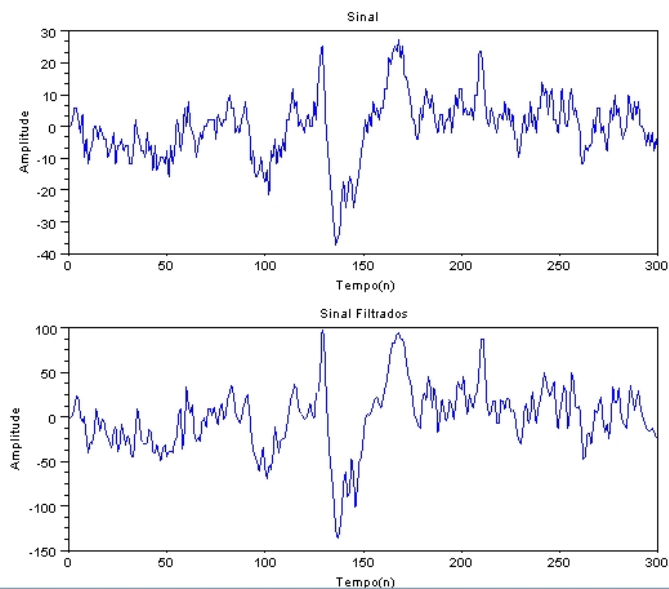
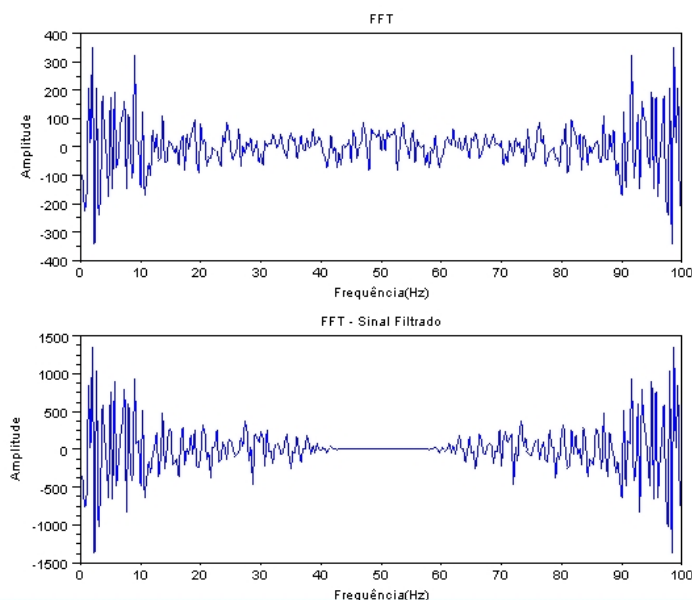


Figura 63 - Sinal sendo filtrado pelo filtro em questão. A cima o sinal natural e abaixo o sinal filtrado.

E aplicando a FFT em cada sinal tem-se o que mostra a Figura 64.





*Figura 64 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. A cima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.*

Visualizando a Figura 62, conclui-se que o filtro em questão se trata de um Butterworth, passa baixa com frequência de corte em aproximadamente 35 Hz. Com isso, observando a Figura 64, comprova-se que o filtro elimina as frequências maiores que 35 Hz, gradativamente, o que comprova o funcionamento da ferramenta de transformação e a integridade dos coeficientes gerados.

#### 6.3.2.3 - Transformação bilinear Z com Chebyshev

Neste exemplo, como dado de entrada são utilizado os polos e zeros de um filtro Chebyshev Tipo I passa baixa. Estes valores foram calculados e inseridos no *Shell* do Scilab conforme a Figura 65.

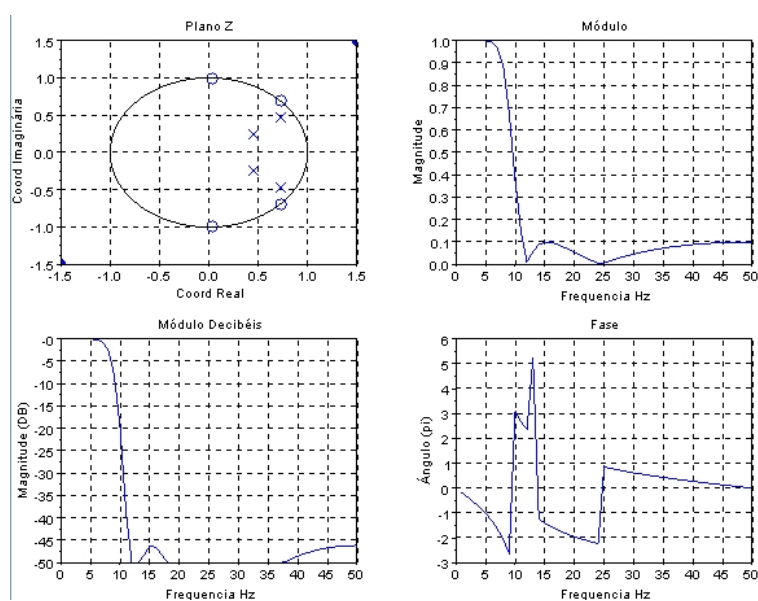
```

96
97 p = [
98 -0.1522 -0.5788070;
99 -0.6837 -0.4460628];
100
101 z = [
102 0 +0.8009j;
103 0 +1.9337j];
104

```

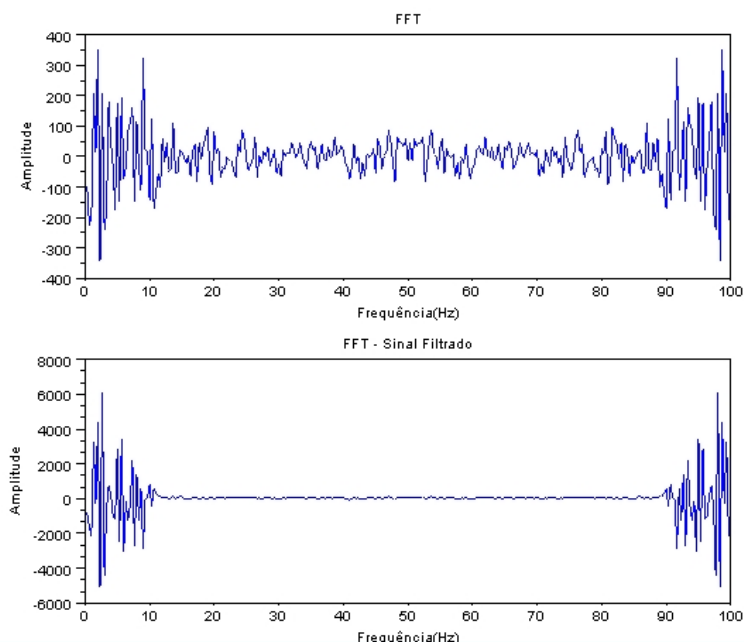
*Figura 65 - Polos e Zeros de um filtro Chebyshev Tipo I Passa Baixa, no plano S.*

Aplicando estes polos e zeros na função de transformação e em seguida nas funções que calculam os coeficientes deste filtro, resultou em um filtro com a configuração conforme mostra Figura 66.



*Figura 66 - Configurações do filtro em questão.*

Um sinal foi escolhido para aplicação do sinal, para que posteriormente a FFT possa ser aplicada. E aplicando a FFT em cada sinal tem-se o que mostra a Figura 67.



*Figura 67 - Gráfico que apresenta a Transformada Rápida de Fourier Discreta. A cima a FFT antes da aplicação do filtro e a baixo a FFT após a aplicação do filtro.*

Visualizando a Figura 66, conclui-se que o filtro em questão se trata de um Chebyshev Tipo I, passa baixa com frequência de corte em aproximadamente 10 Hz. Com isso, observando a Figura 67, comprova-se que o filtro elimina as frequências maiores que 10 Hz, gradativamente, o que comprova o funcionamento da ferramenta de transformação e a integridade dos coeficientes gerados. Também reforçando o método de inserção de polos e zeros, que é reutilizado neste exemplo.

## 6.4 - Avaliação

Seguindo o roteiro proposto pelo *WebMac 4.0 Senior*, o questionário foi aplicado a duas turmas de alunos. Uma delas, graduandos em engenharia elétrica e áreas afins que participaram do segundo mini-curso de Engenharia Biomédica na Prática, a outra foram os alunos do curso Superior de Tecnologia em Sistemas Eletrônicos do IF-SC. Os questionários foram aplicados e de cada turma retirada uma conclusão.

### 6.4.1 - Alunos do Mini-Curso de Engenharia Biomédica na Prática

O questionário foi aplicado para as turmas participantes do segundo Mini-Curso de

Engenharia Biomédica na Prática, e resultou no seguinte gráfico E.S.O.F., conforme Figura 68.

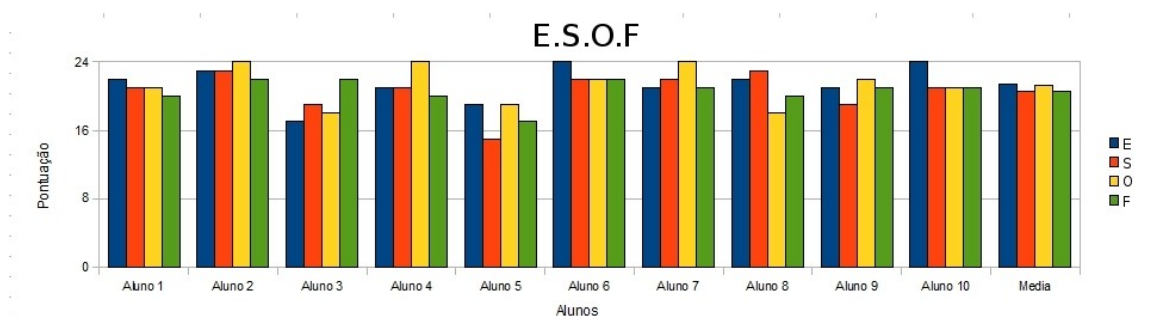


Figura 68 - Gráfico E.S.O.F. do questionário aplicado à turma do segundo Mini-curso de Engenharia Biomédica na Prática. E para Estimulante, S para Significativo, O para Organizado e F para Fácil-de-Usar.

Conforme a Figura 68 e regras de como interpretar os dados do questionário do *Webmac 4.0 Senior*, notamos que em quase todos os quesitos encontraram-se dentro do esperado, que era do fato do sistema ser altamente motivante. A única exceção foi no aluno 5, que segundo ele, no quesito Significativo ainda precisa de algumas melhorias.

Por fim, o gráfico VXS da turma foi plotado, conforme Figura 69.

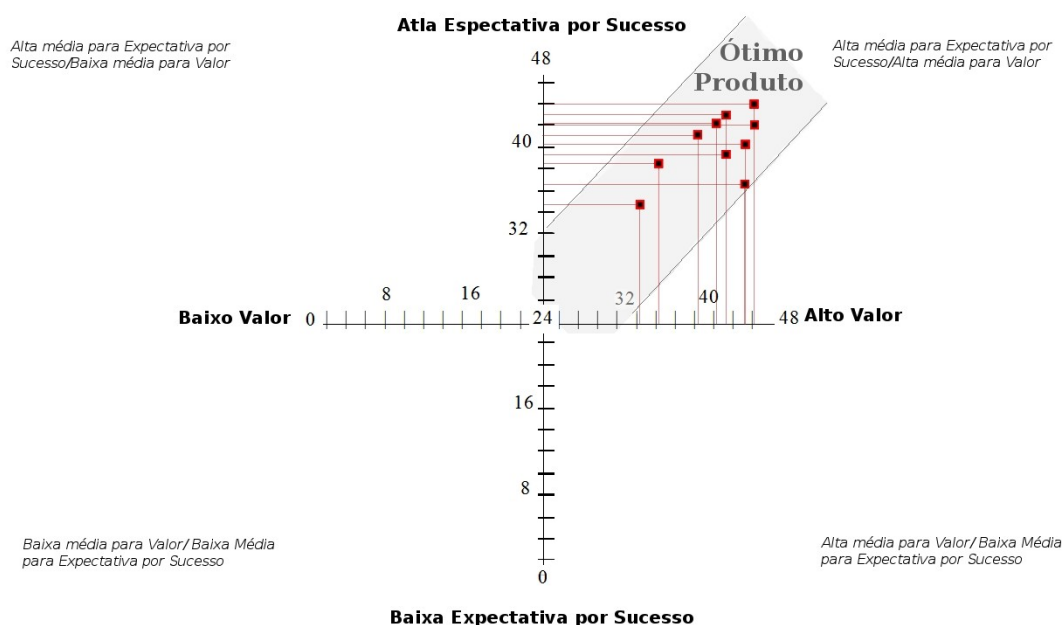


Figura 69 - Gráfico VXS da turma que respondeu o formulário no Segundo Mini-Curso de Engenharia Biomédica na Prática.

Analisando a Figura 69, nota-se que segundo as expectativas propostas pelo *Webmac 4.0 Senior*, os pontos resultantes de cada formulário de cada aluno se localizaram dentro da área proposta como um “WebSite de Sucesso”, neste caso o site do tutorial.

Então, conforme a turma do segundo mini-curso de Engenharia Biomédica na prática, o sítio dos tutoriais de Filtros Digitais é de alto valor e com grande chances de sucesso.

#### 6.4.2 - Alunos de Sistemas Eletrônicos do IF-SC

O questionário foi aplicado para as turmas e resultou no seguinte gráfico E.S.O.F., conforme Figura 70.

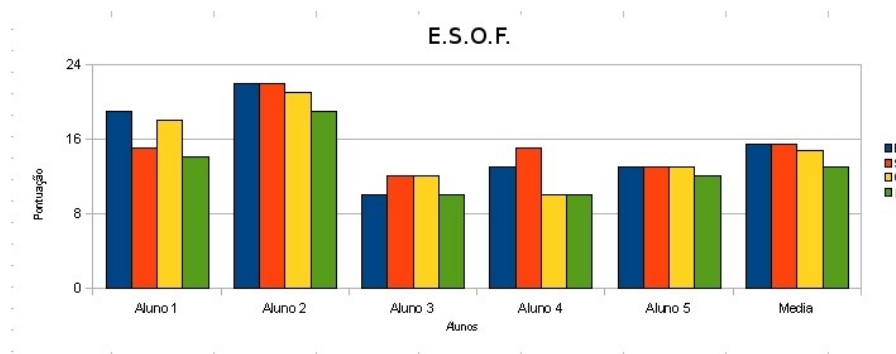


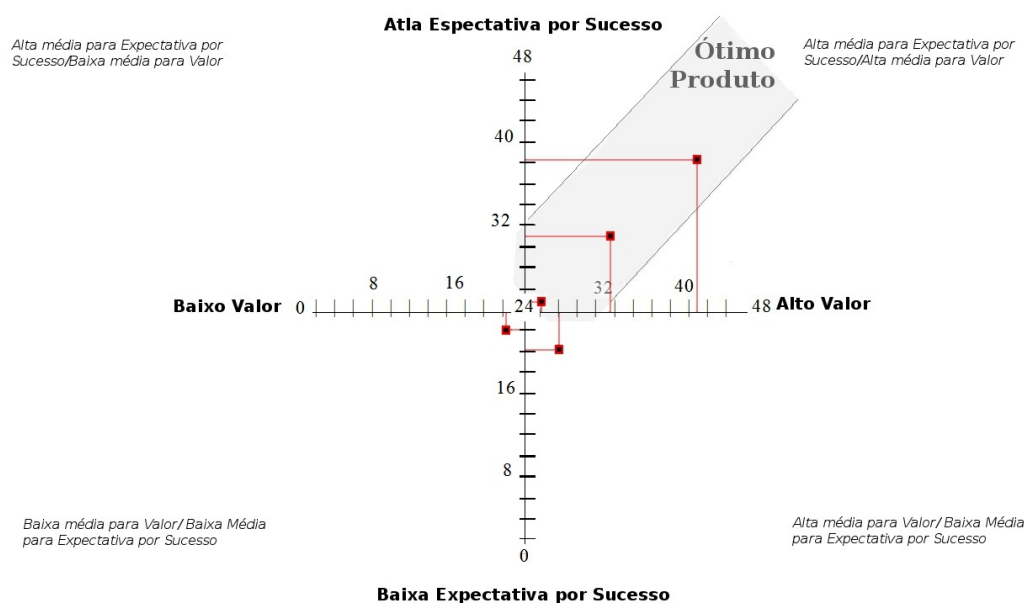
Figura 70 - Gráfico E.S.O.F. do questionário aplicado à turma do IF-SC. E para Estimulante, S para Significativo, O para Organizado e F para Fácil-de-Usar.

Conforme a Figura 70, notamos que em quase todos os quesitos encontraram-se na faixa descrita como “Precisa de Pouco Melhoramento”. Isto ocorreu por se tratar de nível Superior de Tecnologia em Sistemas Eletrônicos, que são pessoas com maior conhecimento na área, o que torna o quesito motivação menor (pois eles já conhecem o conteúdo) e os quesitos técnicos menores (pois estes aluno executam uma visão crítica mais refinada sobre o assunto), encontrando os mínimos problemas do sistema. Esta crítica foi altamente construtiva, pois possibilitou o refinamento das ferramentas implementadas e a correção de diversos *bugs* que dificilmente seriam encontrados por testes tradicionais.

O resultado desta análise, por sua vez não é negativo, pois o fato de o sistema não ser estimulante para essa turma não é tão crítico, uma vez que os tutoriais são desenvolvidos para turma de Engenharia Elétrica, o importante é que esse assunto se torne motivante especialmente para eles.

Esta diferença pode ocorrer porque os alunos de Sistemas Eletrônicos do IF-SC abordam o conteúdo de Processamento Digital de Sinais em todo o seu curso, com muitas aulas práticas. Os alunos de Engenharia Elétrica geralmente possuem apenas uma parte de todo curso esse assunto, o que faz com que os tutoriais sejam mais estimulantes.

Por fim, o gráfico VXS da turma foi plotado, conforme Figura 71.



*Figura 71 - Gráfico VXS da turma que respondeu o formulário do IF-SC.*

Analisando a Figura 71, nota-se que apesar de nem todos os pontos encontrarem-se dentro da faixa esperada, a maioria deles encontram-se dentro da área que avalia o produto como um produto com “Alto valor e alta expectativa de sucesso”.

Então, conforme a turma do IF-SC, o site dos tutoriais de Filtros Digitais, é na maioria, de alto valor e com grande chances de sucesso.

## 6.5 - Resultados

Analisando as Figura 69 e 71, podemos tirar as conclusões sobre o resultado da aplicação dos questionário *Webmac 4.0 Senior*.

Ao observar os resultados do ponto de vista de estudantes do curso de Sistemas Eletrônicos do IF-SC, mostrado na Figura 71, o produto tem um resultado variável, com tendência positiva.

Segundo a maioria dos alunos, o trabalho precisa de pouca melhora em todas as áreas: estimulante, significativo, organização e facilidade no uso.

O tópico estimulante e significativo pode não ter sido em sua pontuação máxima pelo fato de que os alunos de Sistemas Eletrônicos já contêm um alto conhecimento nesta área, sendo assim, estudar novamente algo não ser tão empolgante. Os tópicos facilidade de uso e organização podem ter sido referente a termos técnicos e organização da informação, na qual os tutoriais são levemente diferenciados da maneira como tradicionalmente os professores lecionam processamento digital de dinais, e em si, filtro digitais.

Porém, a idéia inicial era de que estes tutoriais fossem aplicados para alunos de Engenharia Elétrica. Mas, esta aplicação com a turma do IF-SC, garantiu uma visão mais crítica dos tutoriais, que permitiram um refinamento na qualidade da informação.

Já os resultados dos questionários aplicados aos alunos do Mini-Curso de Engenharia Biomédica na Prática, na Figura 69, resultaram em um produto de alto valor e alta chance de sucesso, pois os tutoriais foram projetados especificamente para os alunos de Engenharia Elétrica e cursos afins.

Analisando a Figura 68, nota-se que apenas um aluno julgou um dos tópicos que precisa de pouca melhora. Analisando mais a fundo o questionário do aluno, ele informa que já possuía conhecimento na área. Este fato pode implicar que o aluno julgue o conteúdo não tão significativo pois ele já o conhecia, o que reforça a ideia de que os alunos do IF-SC também já possuíam este conhecimento.

## 7 - CONCLUSÃO

### 7.1 - Conclusão

Quando a leitura de um conteúdo se torna motivante e estimulante, probabilidade de o leitor permanecer atento ao conteúdo é maior. Quando um texto apresenta maneiras de abordar o conteúdo na prática, possibilita que o leitor não precise imaginar muito como as coisas funcionam, mas sim, que foque sua atenção em resultados visíveis e reais, em resultado a suas atividades. Quando um texto motivante e estimulante se une a uma aula prática, traz uma possibilidade ainda maior de tornar o assunto penetrante ao leitor.

Processamento digital de sinais é um conteúdo que precisa destes motivadores trabalhando em paralelo. Primeiro, porque é um assunto complicado de ser compreendido, então há a necessidade de ser motivante e estimulante, pois se o leitor cansar no meio da leitura, ele poderá abandoná-la. Desta maneira, um ambiente amigável e uma escrita leve (semelhante a uma conversa) atraem a atenção do leitor.

Também destacando a utilização prática durante uma leitura, faz com que o leitor não precise imaginar cada resultado ou observar um resultado fixo em uma Figura. Desta maneira a inclusão de JAVA Applets possibilita que mesmo durante a leitura teórica, o Applet possa ir passo-a-passo aplicando as teorias, de maneira que as entradas do leitor não são estáticas, e sim dinâmicas, da mesma maneira que os mais pró-ativos possam entrar com valores diferentes dos do material escrito e visualizar os resultados gerados pela sua curiosidade.

A abordagem prática, como o 'Mão na Massa' permite uma abordagem manual sobre o funcionamento das coisas. Desta maneira, o leitor pode participar interativamente de uma aula com prática com apenas a necessidade de algumas ferramentas, neste caso o Scilab. O Scilab é uma ferramenta de uso gratuito, o que possibilita que o leitor, em qualquer plataforma que suporte este *software* possa realizar esta aula prática, que geralmente (antigamente) poderia ser realizada apenas em laboratórios específicos. Essa abordagem manual do funcionamento, também possibilita que o leitor interaja dinamicamente com o funcionamento do que ele está aprendendo e finalmente visualize o resultado de suas experiências.

Analisando os alunos de Engenharia Elétrica e áreas afins, nota-se a escassez da abordagem sobre processamento digital de sinais (e quando abordado, em pouco volume), o que torna uma abordagem teórica e prática, dinâmica e flexível, do assunto um produto de alta aplicação, pois eles



se sentem motivados e estimulados ao trabalhar com este assunto desta maneira.

Já alunos do curso Superior de Tecnologia em Sistemas Eletrônicos, ou seja, profissionais com conhecimento nesta área, podem não se sentir motivados ao ter uma aula como esta, pois já estudam com mais detalhes o assunto.

De acordo com “A inserção da Engenharia Biomédica nos cursos de graduação de Engenharia Elétrica no Brasil (2008)”, os cursos de Engenharia Elétrica e áreas afins apresentam uma escassez nos assuntos que decorrem a Engenharia Biomédica e ainda mais em aulas práticas. Sendo assim, o produto deste trabalho se encaixa adequadamente com o proposto, que era o de um conteúdo teórico e prático, ao mesmo tempo estimulante e motivante, para alunos de Engenharia Elétrica.

Com a aplicação deste trabalho, os alunos de Engenharia Elétrica terão uma abordagem sobre filtragem digital de sinais com foco em Engenharia Biomédica, que ajuda a fechar a lacuna da falta da abordagem deste conteúdo na graduação e também ajuda a fechar a lacuna da falta de abordagem sobre Engenharia Biomédica, tanto na teoria como na prática.

## 7.2 - Trabalhos Futuros

Este trabalho traz algumas possibilidades, além de diversas outras que já são possíveis com a Plataforma Didática de Filtragem de Sinais Biomédicos.

Uma das possibilidades de expansão seria a criação de um tutorial didático para utilização de filtros digitais em *hardware*. Uma vez que este trabalho já apresenta os projetos de filtros digitais e ensina a utilizar estes filtros em *software*, a criação de um módulo para a Plataforma Didática de Processamento de Sinais Biomédicos poderia dar fim ao fluxo de conhecimento que aborda a filtragem digital. Por exemplo, o aluno poderia projetar o filtro utilizando este trabalho e posteriormente gravar as informações em um microcontrolador e visualizar o filtro em atividade com dados coletados de um paciente em tempo real, onde esse paciente pode ser um outro aluno que esteja no momento estudando.

Outra possibilidade seria um ramo paralelo, em que o assunto abordado seria outros tipos de processamento digital de sinais utilizados na Engenharia Biomédica. Como, por exemplo, geradores de funções digitais (ECG, EEG, etc), filtros de detecção de momentos (Detector de Espículas, Detector de onde QRS, no ECG), etc.

## ANEXO I – QUESTIONÁRIO WEBMAC 4.0 SENIOR

Questionário para Avaliação da

### Plataforma SPSB – Filtragem Digital

**Curso de Graduação:**

**Semestre:**.....

**Instruções:** Assinale com um **X** a nota mais apropriada para cada afirmação abaixo. O significado de cada nota é o seguinte:

- 3** = Concordo plenamente, com muita ênfase (muito, sempre, etc.)
- 2** = Concordo parcialmente (um pouco, às vezes, etc.)
- 1** = Discordo um pouco, com pouca ênfase (acho que não, quase nunca, etc.)
- 0** = Discordo fortemente, com muita ênfase (absolutamente não, nunca, etc.)
- NA** = Não aplicável.

1O padrão de cores utilizado nos Applets é agradável.	<b>3 2 1 0 NA</b>
2As informações do Tutorial são precisas e equilibradas entre os vários assuntos.	<b>3 2 1 0 NA</b>
3Os recursos visuais (animações, figuras, Applets) incluídas neste Tutorial ajudam a apresentar o tópico.	<b>3 2 1 0 NA</b>
4As imagens dos Tutoriais são nítidas e claras.	<b>3 2 1 0 NA</b>
5Os títulos são chamativos e informativos em cada tópico do Tutorial.	<b>3 2 1 0 NA</b>
6Os roteiros trazem referências de onde mais assuntos sobre o conteúdo pode ser encontrado.	<b>3 2 1 0 NA</b>
7Eu encontrei a quantidade de informação que eu precisava no Tutorial e nos roteiros de aula prática.	<b>3 2 1 0 NA</b>
8O texto referente ao funcionamento dos Applets e roteiros práticos são suficientes para ajudar a utilizar os mesmos.	<b>3 2 1 0 NA</b>
9Esta Plataforma Didática para ensino de Engenharia Biomédica (o tutorial + o roteiro de aula prática + os Applets) estimula o aprendizado.	<b>3 2 1 0 NA</b>
10As informações apresentadas no Tutorial são consistentes (parecem estar bem fundamentadas).	<b>3 2 1 0 NA</b>
11Há um menu, ou sumário no início que me permite saber quais assuntos estão contidos dentro do Tutorial.	<b>3 2 1 0 NA</b>
12Sempre que preciso retornar para uma parte específica do tutorial, consigo fazer com facilidade.	<b>3 2 1 0 NA</b>
13As informações incluídas no Tutorial são interessantes (despertaram entusiasmo).	<b>3 2 1 0 NA</b>
14As informações encontradas nos Tutoriais são correntes e atualizadas.	<b>3 2 1 0 NA</b>
15O propósito desta aula prática está sempre claro para mim.	<b>3 2 1 0 NA</b>
16Os Roteiros de Aula Prática possuem informações suficientes para a realização dos experimentos.	<b>3 2 1 0 NA</b>

17A variedade de formatos (textos e imagens) do Tutorial e dos Roteiros de Aula Prática retém minha atenção.	<b>3 2 1 0 NA</b>
18As informações do Tutorial são úteis para mim.	<b>3 2 1 0 NA</b>
19Todas as informações do Tutorial são apresentadas usando linguagem e estilo claros e consistentes.	<b>3 2 1 0 NA</b>
20Todos os atalhos e <i>links</i> estão funcionando adequadamente.	<b>3 2 1 0 NA</b>
21O roteiro possui características únicas que o faz mais interessante.	<b>3 2 1 0 NA</b>
22A Plataforma SPSB se mostrou útil para o aprendizado sobre filtragem digital de sinais.	<b>3 2 1 0 NA</b>
23O texto do Tutorial é bem escrito, sem erros gramaticais ou outros tipos de erros.	<b>3 2 1 0 NA</b>
24O tempo todo eu posso controlar que informação do Tutorial eu desejo ver.	<b>3 2 1 0 NA</b>
25Há coisas surpreendentes na Plataforma SPSB.	<b>3 2 1 0 NA</b>
26O Sistema fornece oportunidades para interatividade.	<b>3 2 1 0 NA</b>
27As instruções de uso desta Plataforma são simples e claras.	<b>3 2 1 0 NA</b>
28Todos os botões (links) e mecanismos de navegação funcionam adequadamente com sua proposta. (Incluindo a navegação nos Applets)	<b>3 2 1 0 NA</b>
29O <i>Layout</i> dos Applets são atrativos.	<b>3 2 1 0 NA</b>
30Existem somente informações importantes ou relevantes no Tutorial.	<b>3 2 1 0 NA</b>
31Não importa onde eu esteja no Tutorial, eu posso retornar para a página inicial ou sair.	<b>3 2 1 0 NA</b>
32O tempo previsto para realizar as experiências das aulas práticas é suficiente.	<b>3 2 1 0 NA</b>

Antes de responder as próximas perguntas, volte a cada afirmação em que você assinalou **NA** (não aplicável) e anote uma pontuação baseado no seguinte critério:

**0** ponto se a Plataforma SPSB seria beneficiada se tivesse incluído este item.

**1** ponto se a Plataforma SPSB não necessita deste item.

**2** pontos se a Plataforma SPSB ficou melhor sem a inclusão deste item.

É a primeira vez que estuda estes conteúdos? (**NÃO/SIM**)

Qual é a melhor coisa sobre a Plataforma SBSP? \_\_\_\_\_

\_\_\_\_\_  
 \_\_\_\_\_

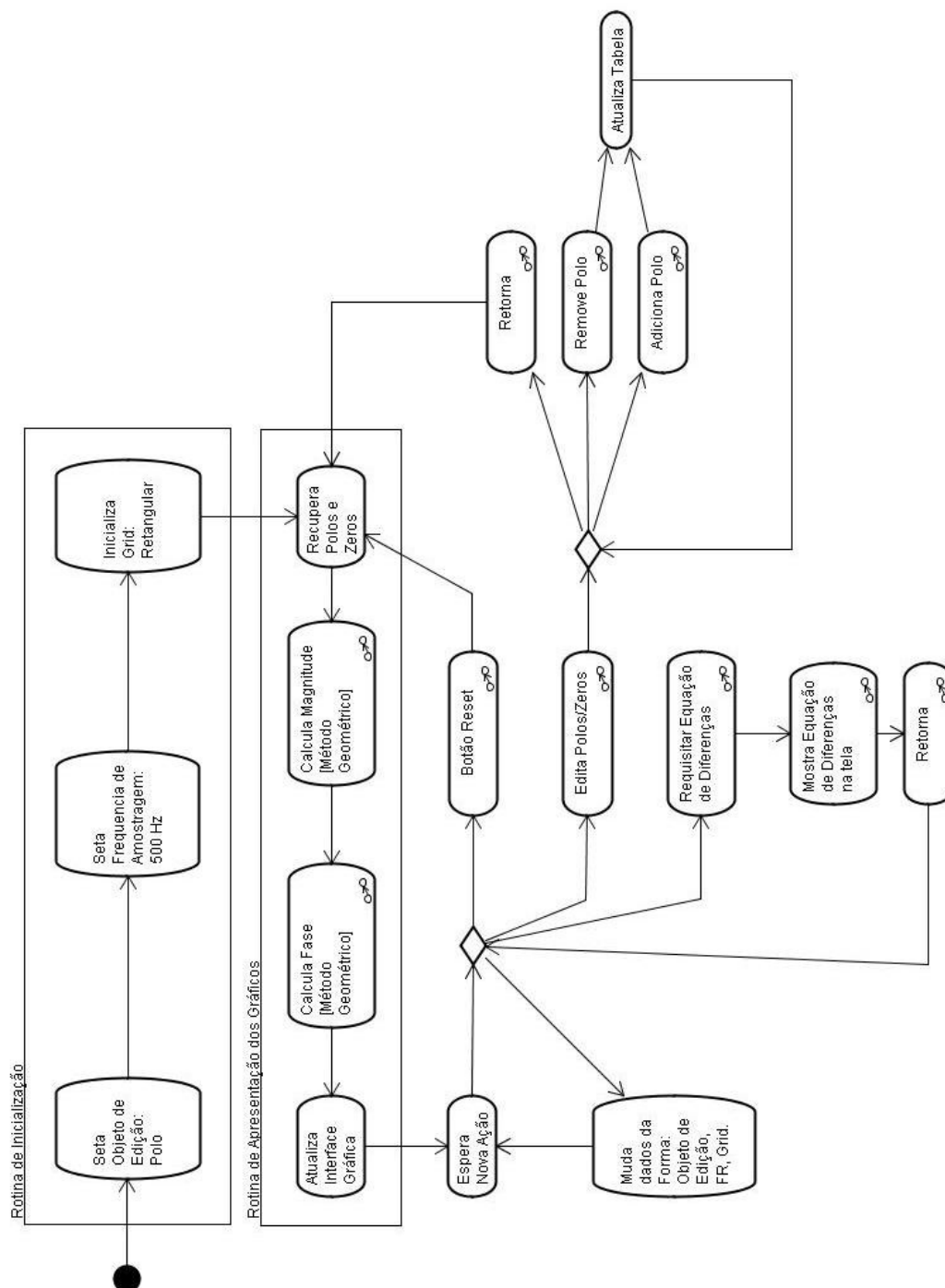
O que precisa de melhoria na Plataforma SPSB? \_\_\_\_\_

\_\_\_\_\_  
 \_\_\_\_\_

Agradecemos por sua contribuição neste trabalho.

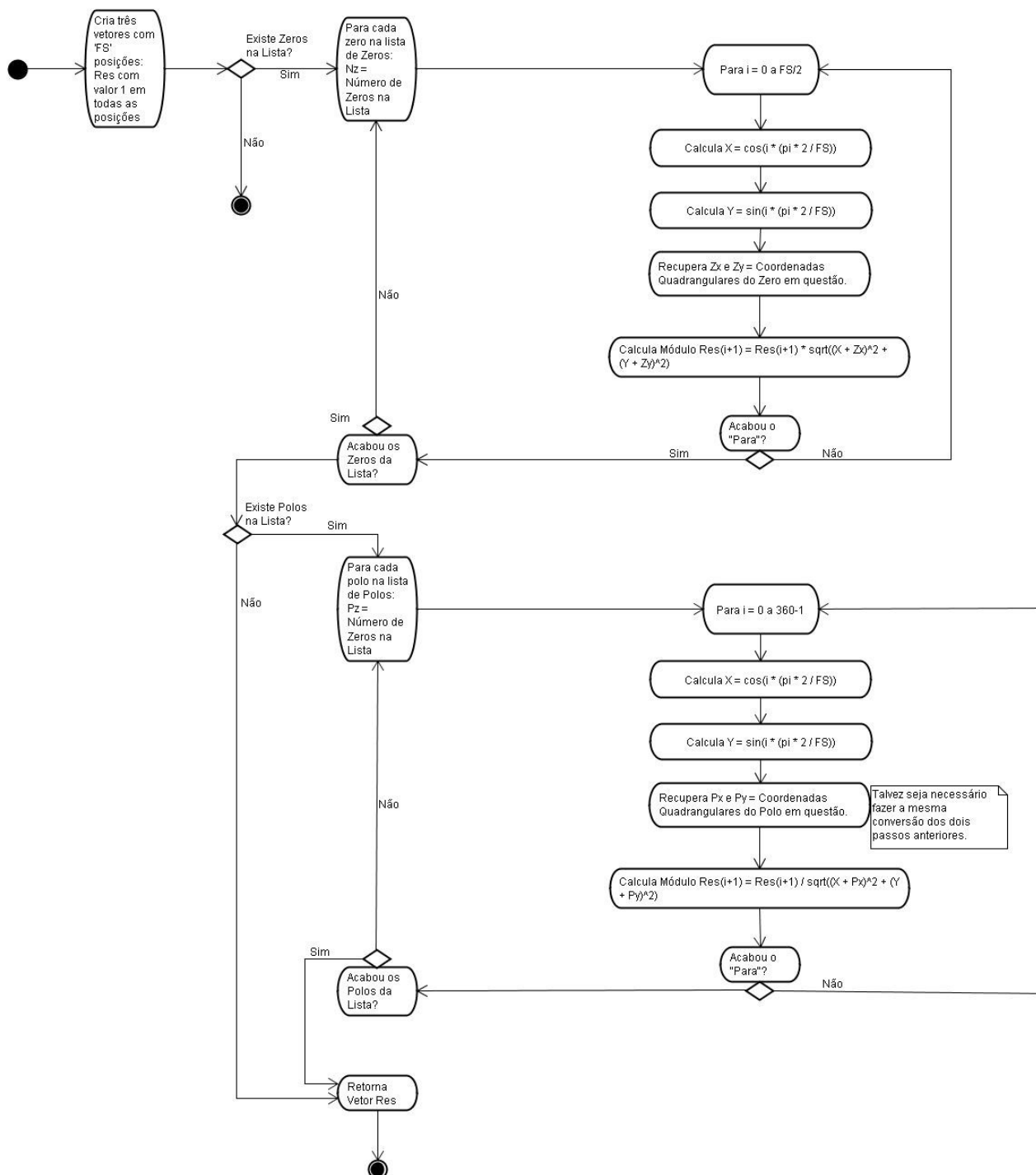
## APÊNDICE I – DIAGRAMAS DE MODELAGEM

Diagrama de Atividades do PlanoZ





### Diagrama de Atividades do Calcula Magnitude



## Diagrama de Atividades do Calcula Fase

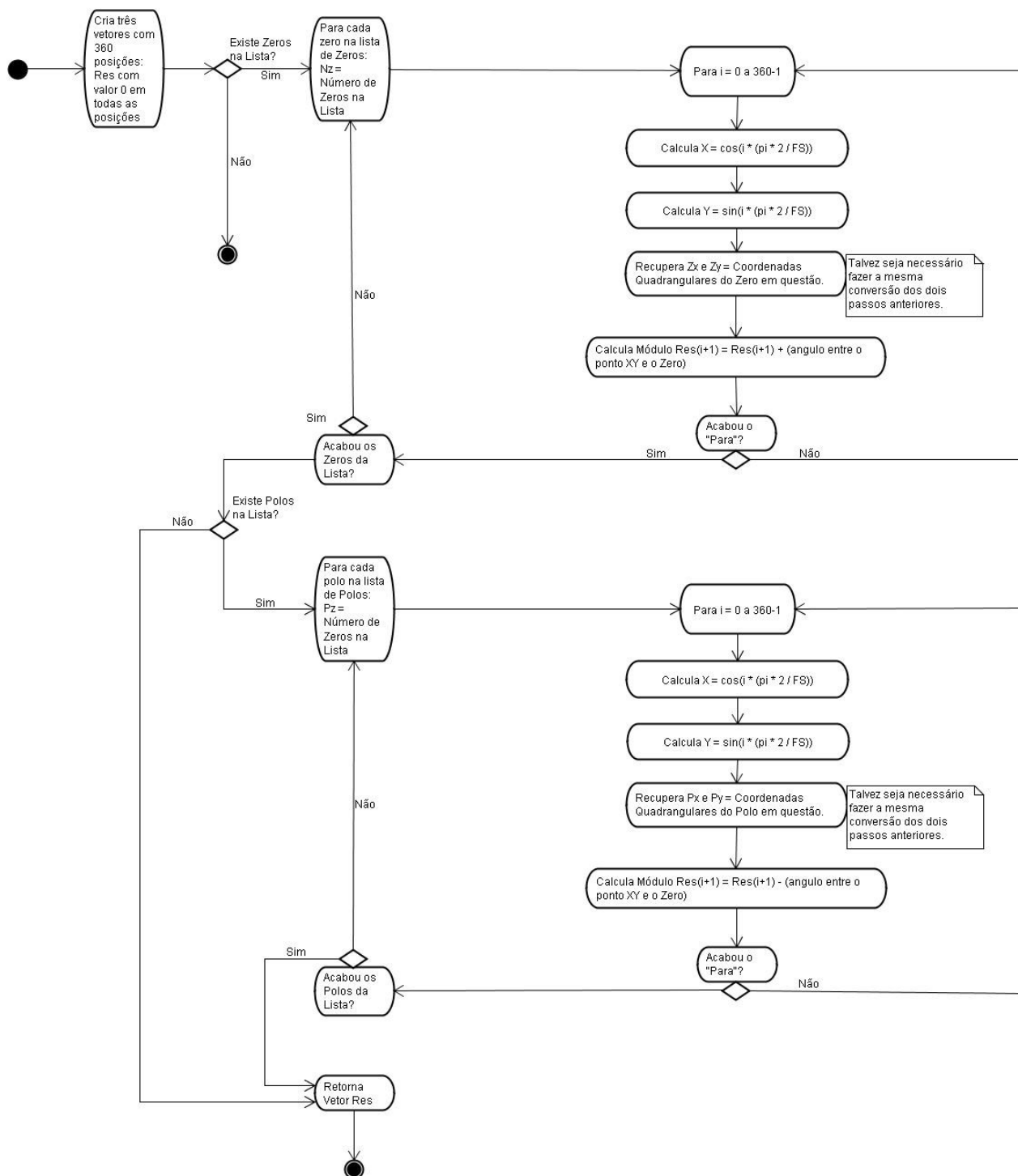


Diagrama de Atividades do FiltroFIR

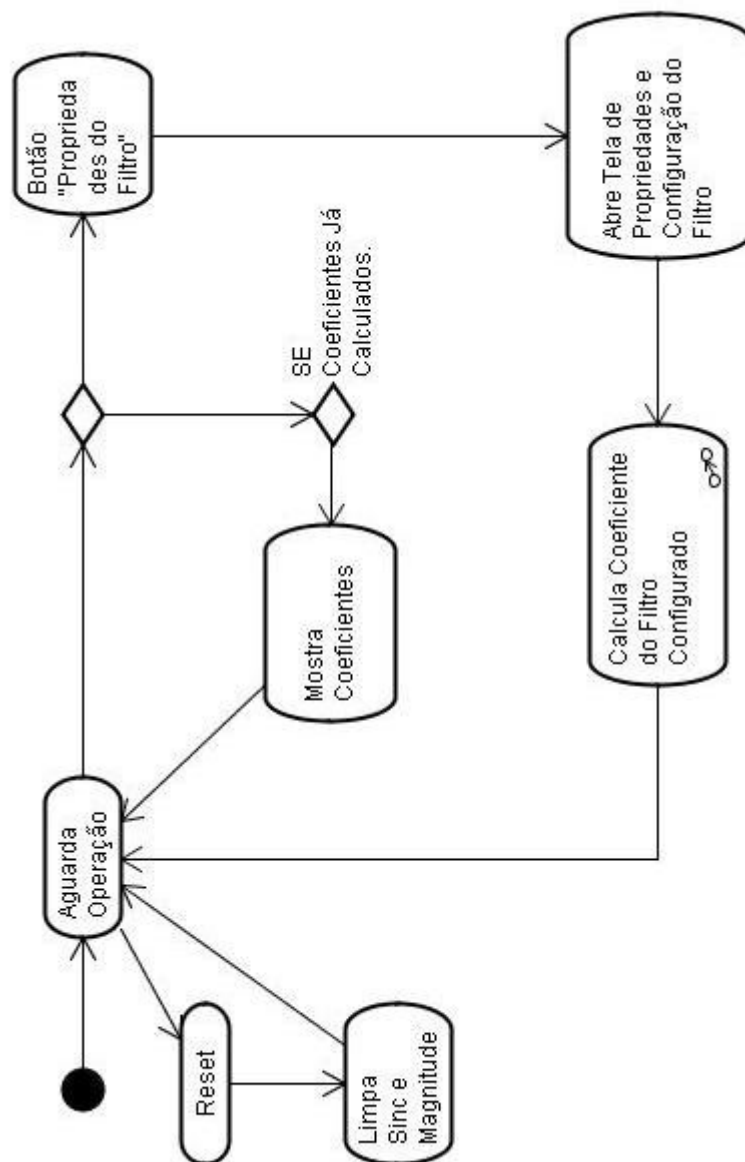
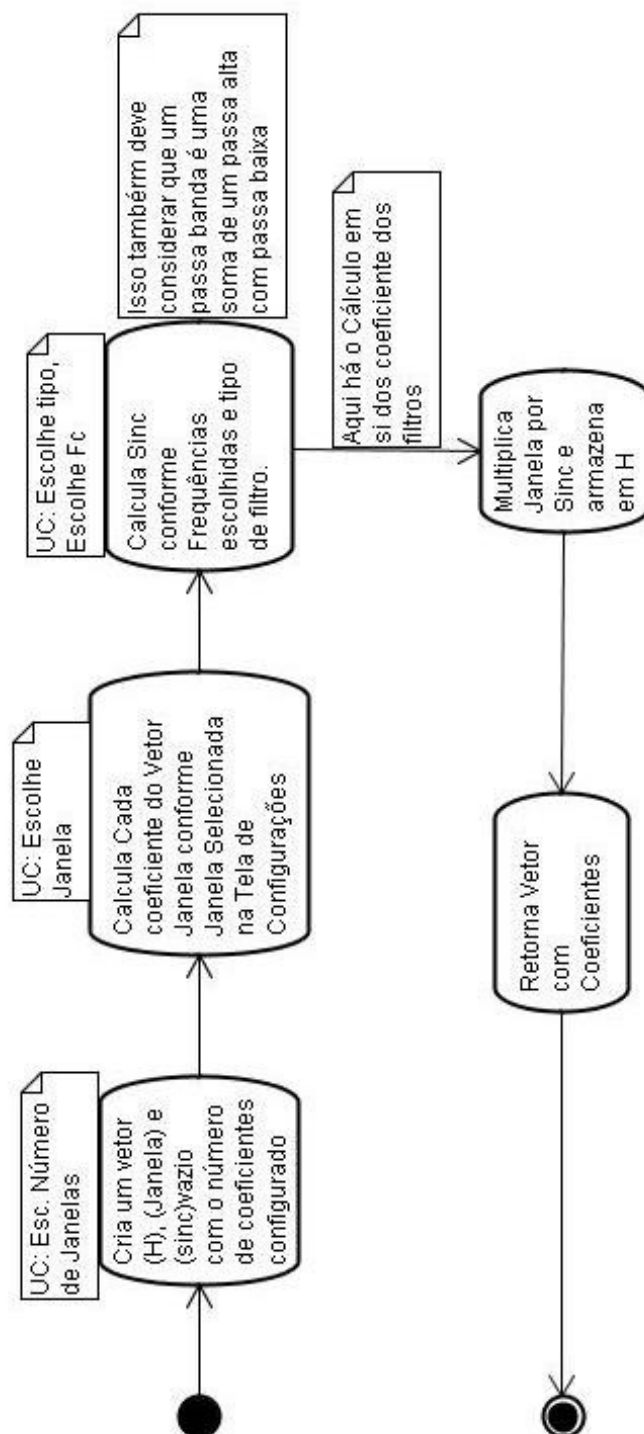




Diagrama de Atividades do caso “Calcula Coeficientes”



### Diagrama de Classe do FiltroFIR

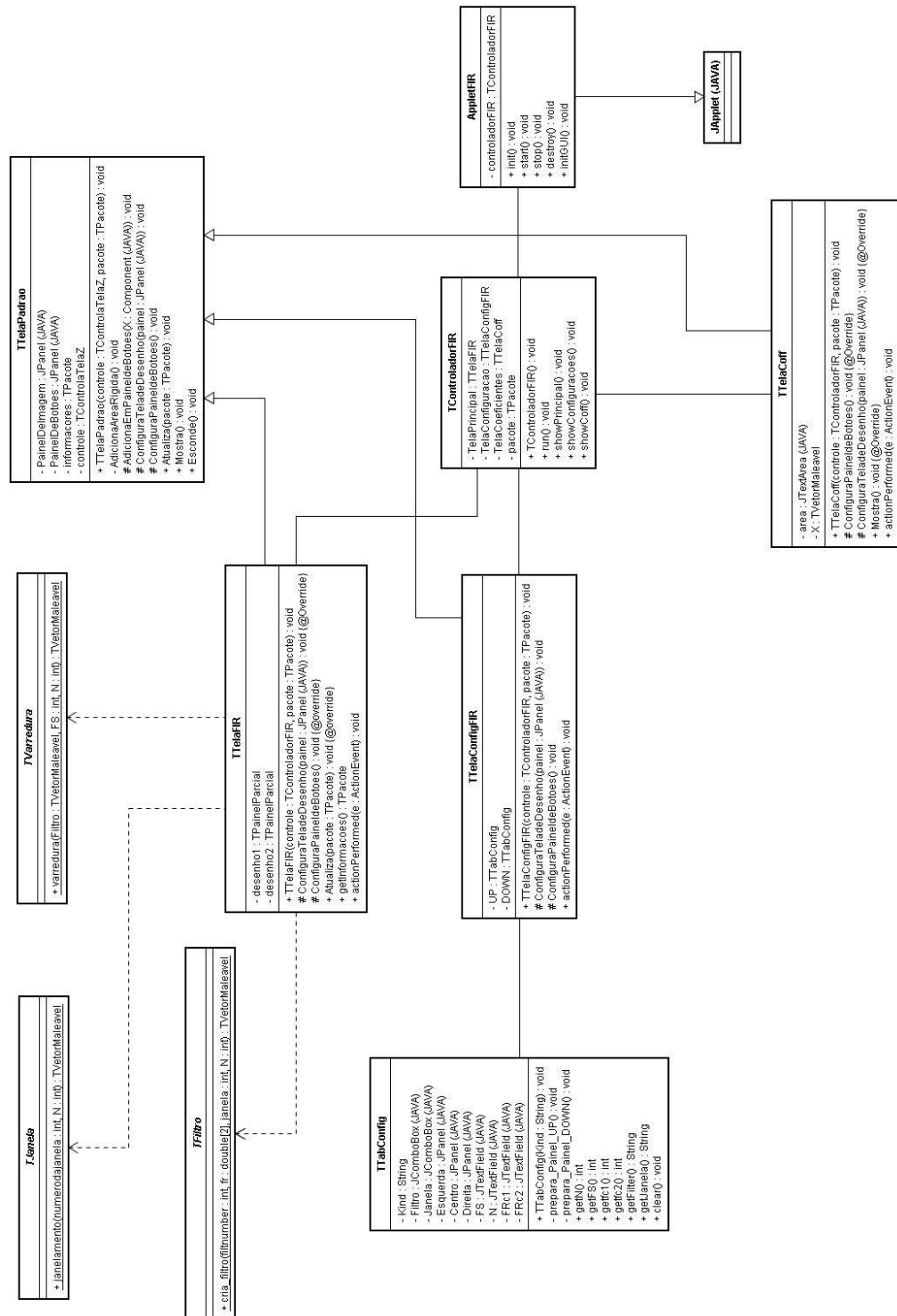


Diagrama de Atividades do BZT

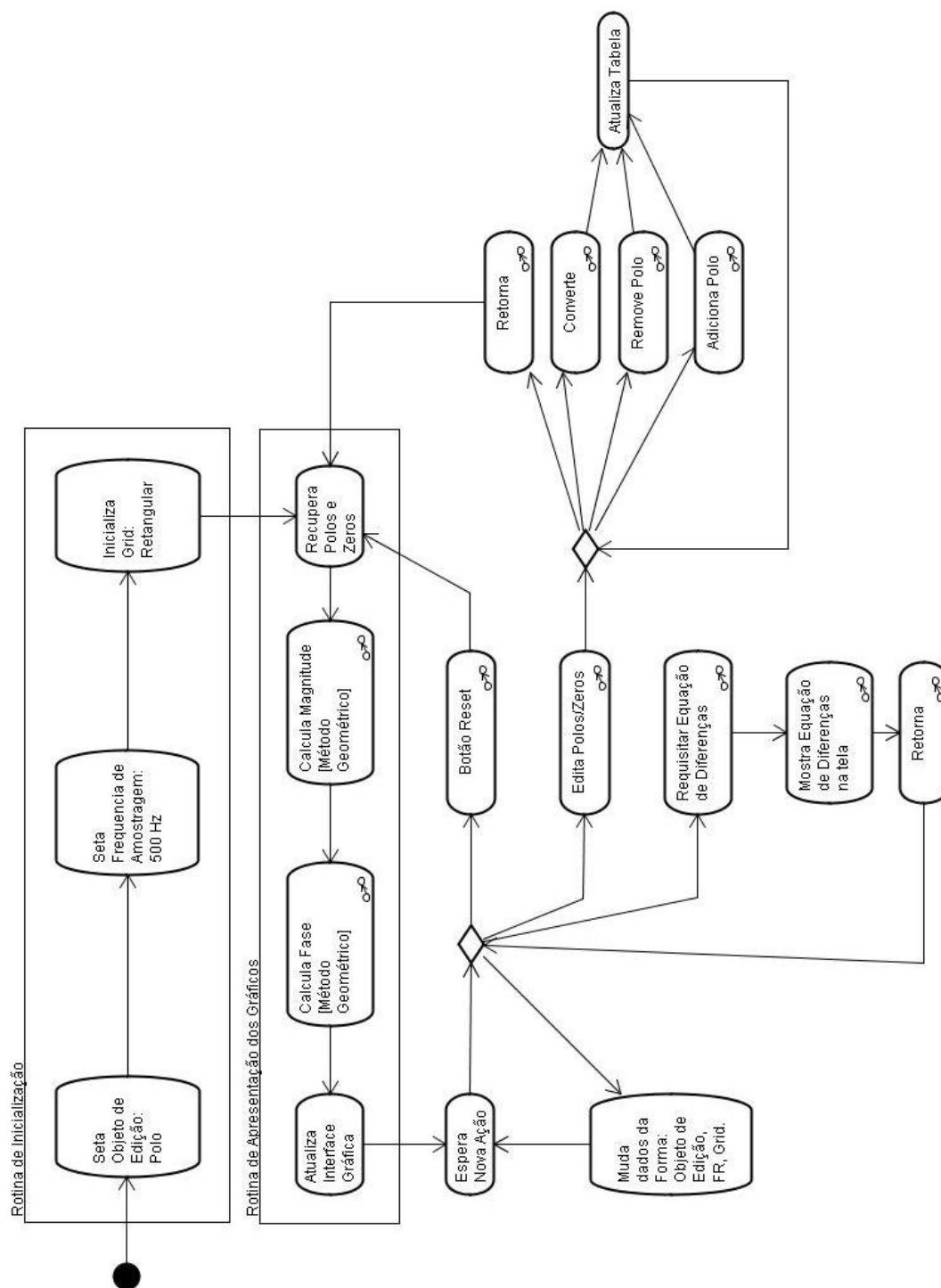
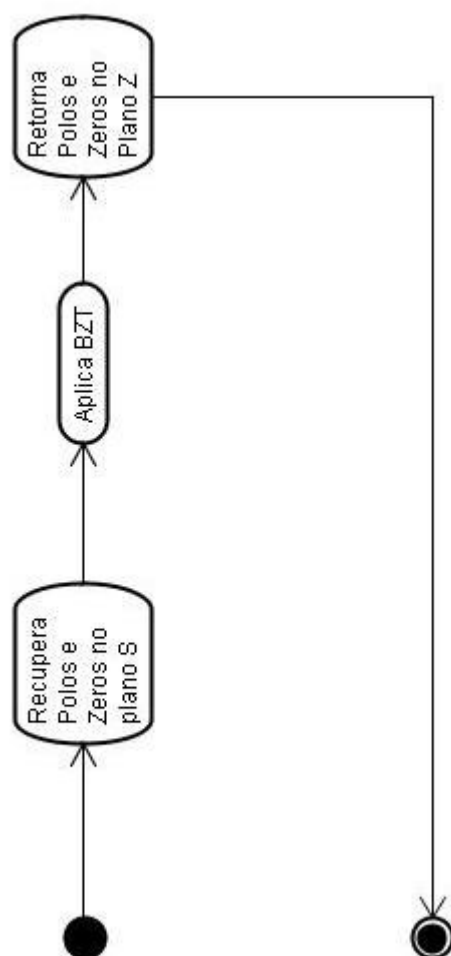
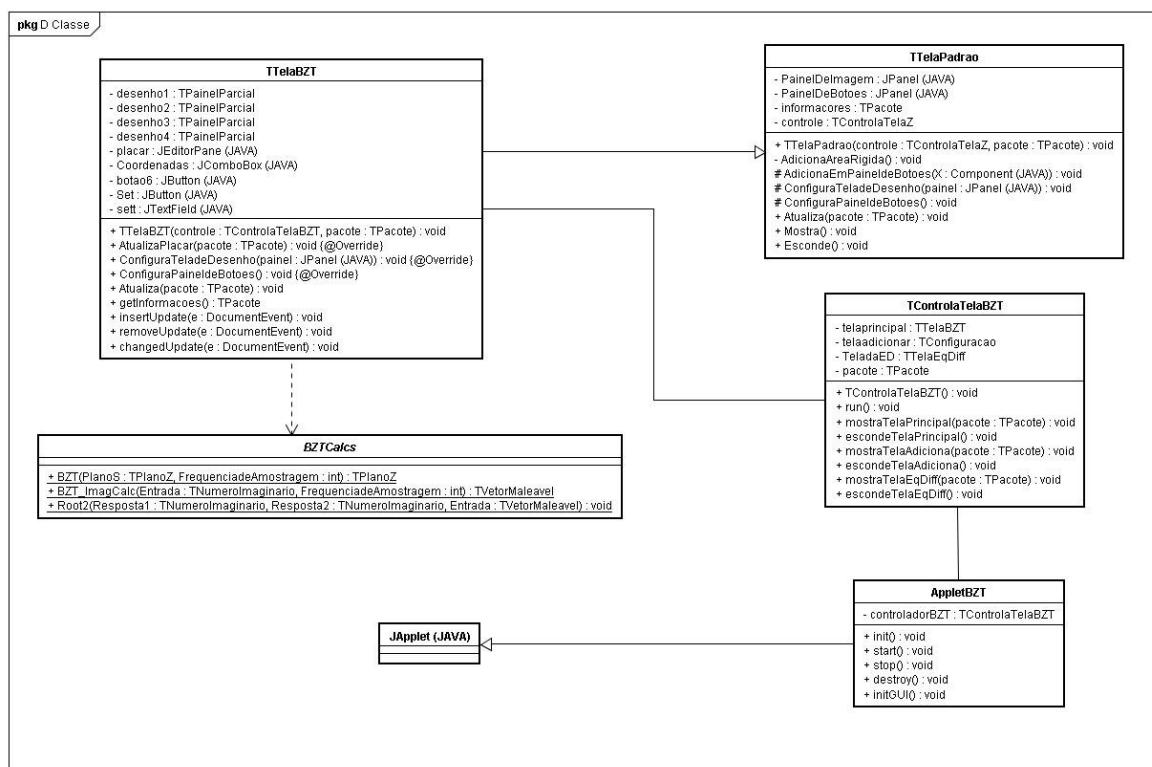


Diagrama de Atividades da Atividade “Converte”



## Diagrama de Classe do BZT



## Diagrama de Atividades do FILTRO

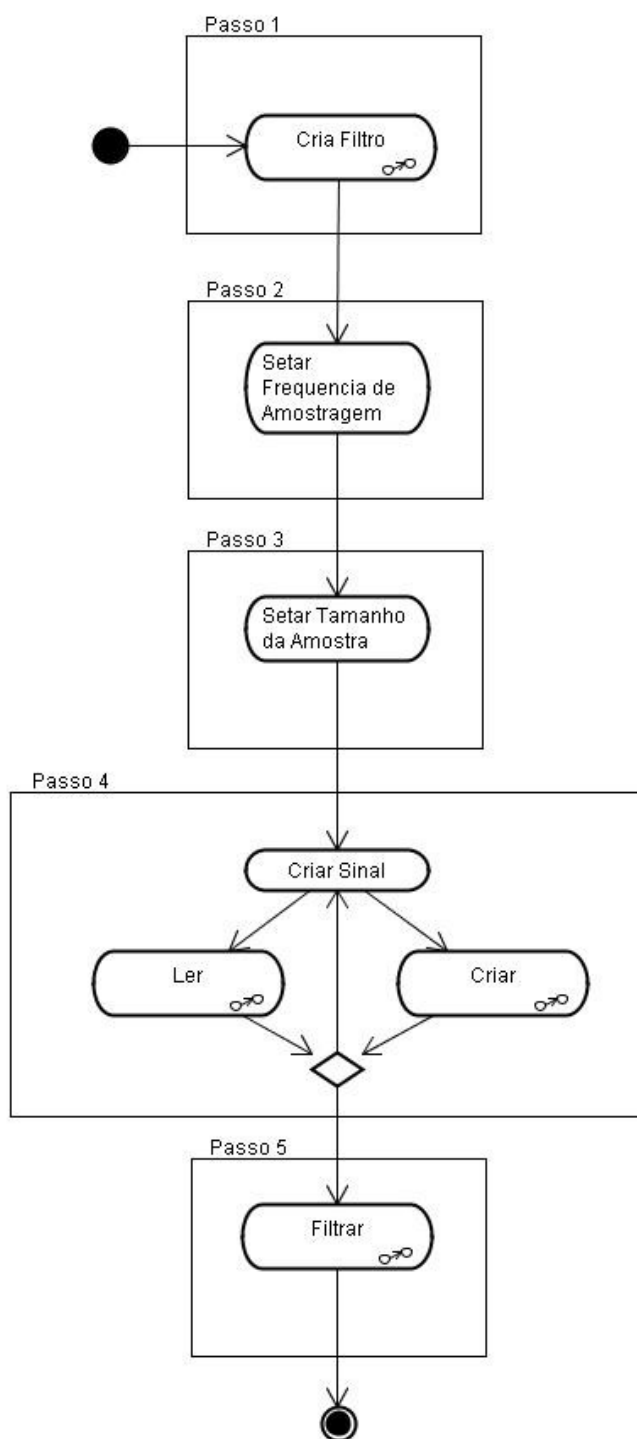


Diagrama de Atividade da Atividade Ler

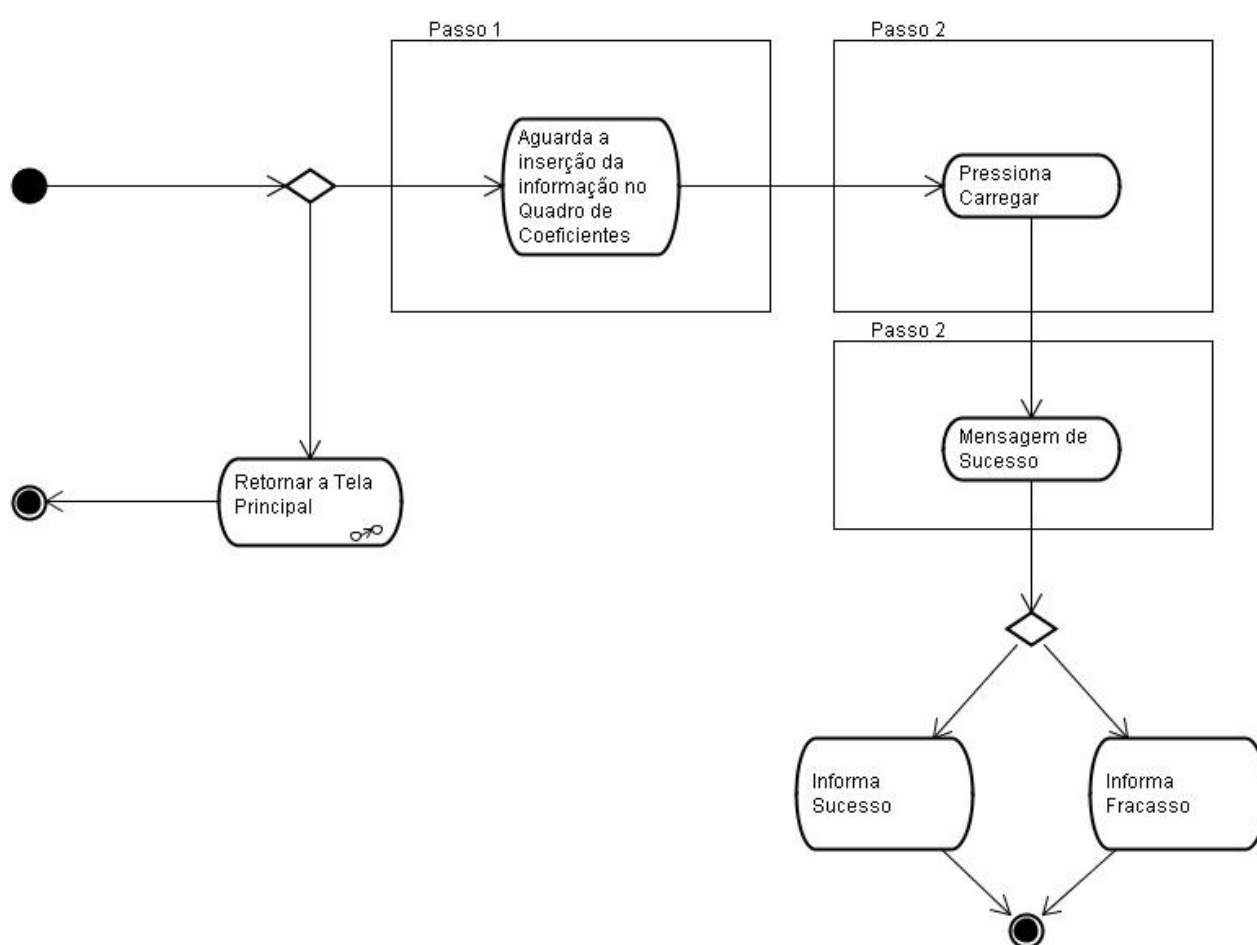


Diagrama de Atividade da Atividade Cria Filtro

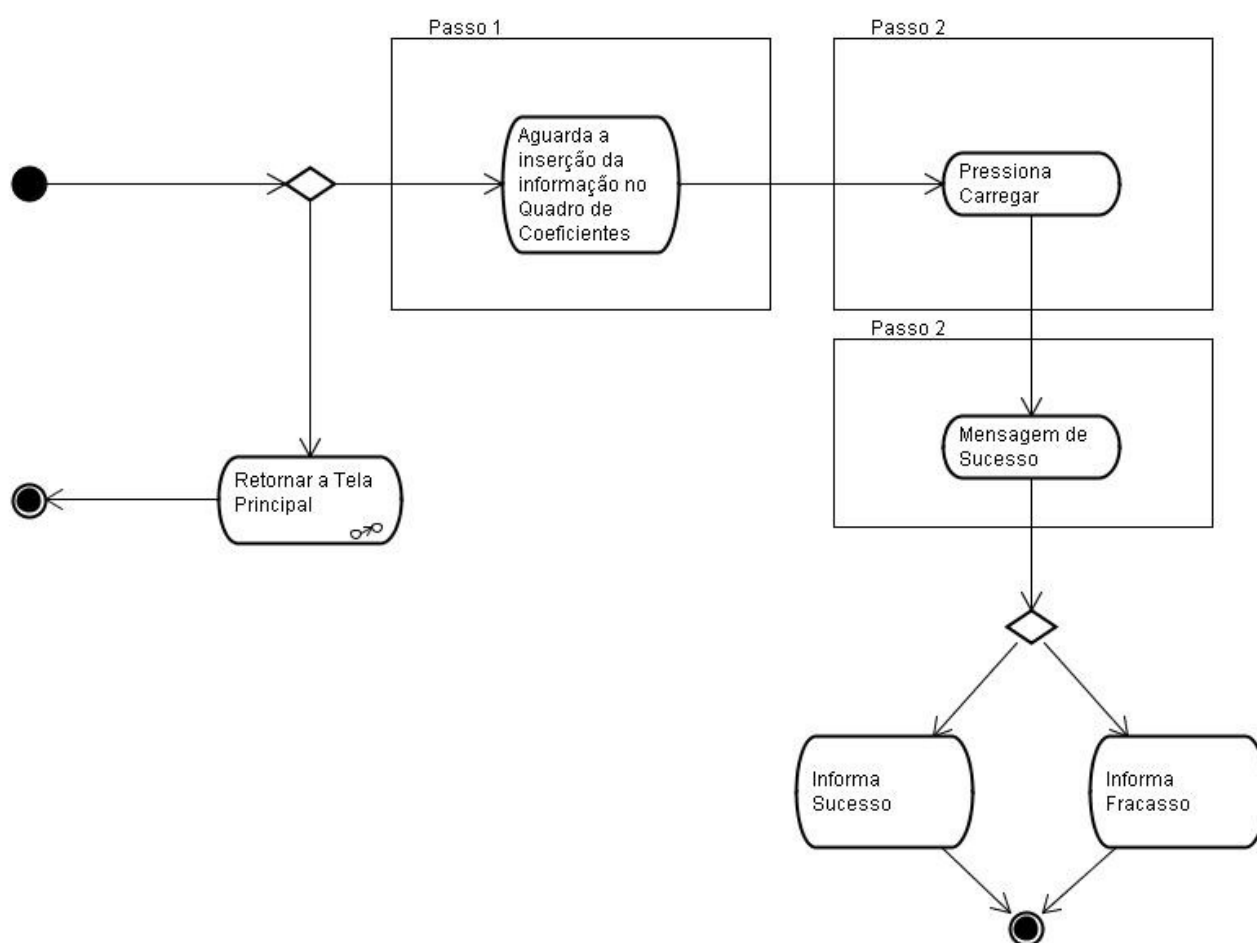
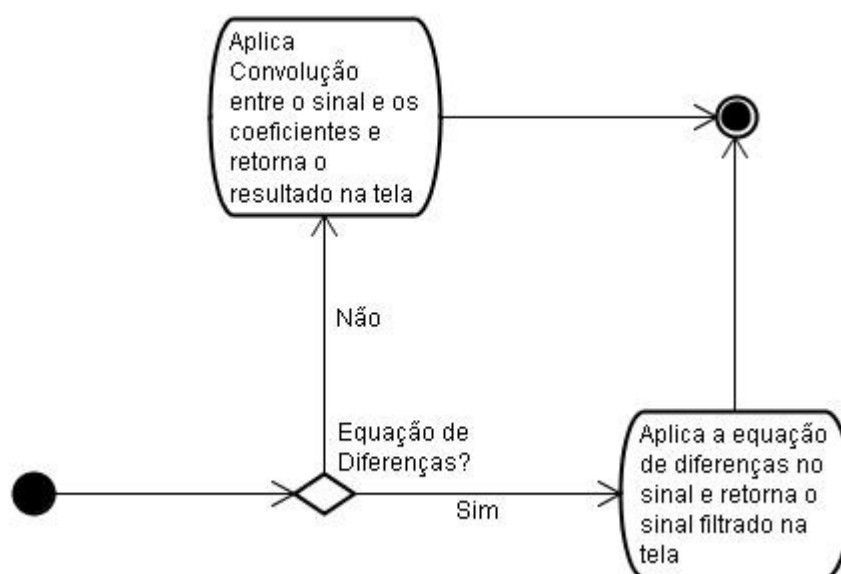
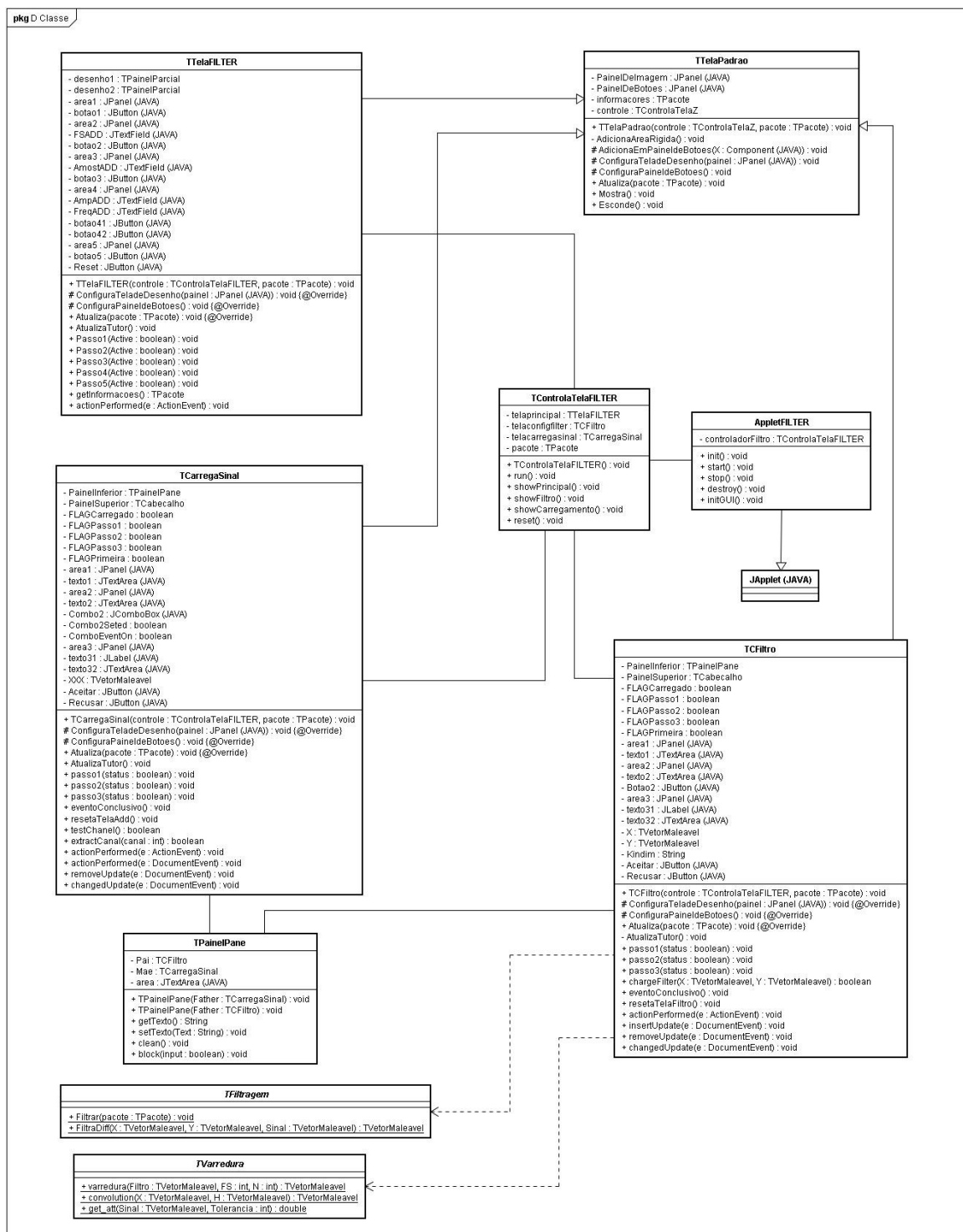




Diagrama de Atividade da Atividade Filtrar



## Diagrama de Classe do FILTRO



## APÊNDICE II – TEXTO DE AJUDA DAS FUNÇÕES DO SCILAB

### **eq\_diff = Equação Diferenças**

Sequencia de Chamada:

**eq = eq\_diff(Z,P)**

Parâmetros:

Z = Zeros do Plano.

Ex: = [0.5 0.6], onde o 0.5 é a parte real e 0.6 é a parte imaginária.

PS: Não deve ser utilizado os números imaginário do SciLab.

P = Polos do Plano.

PS: Da mesma maneira do “Z”.

eq = Equação de diferenças.

A resposta se dá em duas linhas, na primeira contendo os coeficientes dependentes de X e na segunda linha os dependentes de Y. A ordem se dá na ordem da matriz, sendo a primeira coluna o  $Z^{-1}$ , segunda coluna o  $Z^{-2}$  e assim por diante.

Descrição:

Calcula a equação de diferenças de um sistema discreto a partir dos polos e dos zeros do sistema.

Exemplos:

```
Z = [0 1]
P = [0 0.9]
eq = exp_calc(Z,P);
```

Autor: Diego Laucsen da Rosa

**exp\_calc – Calculo de Expressões**Sequencia de Chamada:

```
equ_solve = exp_calc(a,b);
```

Parâmetros:

a,b: Expressão polinomial no modo vetor.

Ex: [1 1 5], que é igual a  $1x^2 + 1X + 5$

equ\_solve: Resultado da multiplicação polinomial entre os dois polinômios, a e b.

Descrição:

Calcula a multiplicação polinomial.

Exemplos:

```
a = [1 1]
```

```
b = [5 9 7]
```

```
equ_solve = exp_calc(a,b);
```

Veja Também:

plota\_tela\_z e eq\_diff

Autor: Diego Laucsen da Rosa

**Filtra – Filtra sinais com equação de Diferenças.**Sequencia de Chamada:

```
sinal_filtrado = filtra(sinal,filtro)
```

Parâmetros:

sinal: Sinal que se deseja filtrar carregado em um vetor.

filtro: a equação de diferenças gerado pela função: eq\_diff

sinal\_filtrado: O Sinal devidamente filtrado.

Descrição:

Filtra um sinal a partir de uma equação de diferenças.

Exemplos:

```
sinal = [0 0 0 1 0 0 0]
filtro = [1 -1; 1 -0.9]
sinal_filtrado = filtra(sinal,filtro)
plot(sinal_filtrado)
```

Veja Também:

eq\_diff

Autor: Diego Laucsen da Rosa

**geo\_ang\_z = Ângulo entre dois Pontos no plano Z**

Sequencia de Chamada:

```
angulo = geo_ang_z(ponto1,ponto2);
```

Parâmetros:

ponto1: Um dos pontos a ser medido. É dado como preferência o ponto de referência.

ponto2: Ponto a ser comparado com o ponto1. Este por sua vez é variável.

angulo: Ângulo entre os dois pontos.

Descrição:

Calcula o ângulo gerado entre a reta formada pelos dois pontos ponto1 e ponto2 em relação ao eixo X.

Exemplos:

```
P1 = [ 0 1];
```

```
P2 = [5 6];
```

```
distancia = geo_ang_z(P1,P2)
```

Veja Também:

geo\_teta\_z

Autor: Diego Laucsen da Rosa

**geo\_dist\_z = Distância Geométrica para o Plano Z**Sequencia de Chamada:

```
distancia = geo_dist_z(ponto1,ponto2);
```

Parâmetros:

referência.  
variável.

ponto1: Um dos pontos a ser medido. É usado como ponto de  
ponto2: Ponto a ser comparado com o ponto1. Este por sua vez é  
distancia: Distância entre os dois pontos.

Descrição:

Calcula a distancia entre dois pontos, como no plano cartesiano.

Exemplos:

```
P1 = [ 0 1];  
P2 = [5 6];  
distancia = geo_dist_z(P1,P2)
```

Veja Também:

geo\_modulo\_z

Autor: Diego Laucsen da Rosa

**geo\_modulo\_z = Módulo (Magnitude) Geométrico do plano Z**Sequencia de Chamada:

```
vector_modulo = geo_modulo_z(zeros_in,polos_in,fs);
```

Parâmetros:

polos\_in: Lista de Polos. Ex:  $P = [a \ b; \ c \ d]$ , onde  $a$  e  $c$  são a parte real e  $b$  e  $d$  a imaginária.

zeros\_in: Lista de Zeros. Ex. No mesmo modelo dos Polos.

Fs: Frequência de Amostragem.

vector\_modulo: Vetor de qualquer onde se encontrará um vetor que poderá ser plotado para visualização da resposta.

Descrição:

Calcula a resposta em frequência (magnitude) de um sistema a partir dos Polos e dos Zeros do sistema no domínio do plano Z.

Exemplos:

```
Z = [ 0  1,  0 -1];
```

```
P = [0 +0.95, 0 -0.95];
```

```
vector_modulo = geo_modulo_z(zeros_in,polos_in,240);  
plot(vector_modulo);
```

Veja Também:

plota\_tela\_Z, geo\_teta\_z e eq\_diff

Autor: Diego Laucsen da Rosa



**geo\_teta\_z = Teta (Fase) Geométrico do plano Z**Sequencia de Chamada:

```
vector_teta = geo_teta_z(zeros_in, polos_in, fs);
```

Parâmetros:

polos\_in: Lista de Polos. Ex:  $P = [a \ b; \ c \ d]$ , *onde a e c são a parte real e b e d a imaginária.*

zeros\_in: Lista de Zeros. Ex. No mesmo modelo dos Polos.

Fs: Frequência de Amostragem.

vector\_teta: Vetor de qualquer onde se encontrará a resposta da função

Descrição:

Calcula a defasagem de um sistema a partir dos Polos e dos Zeros do sistema no domínio do plano Z.

Exemplos:

```
Z = [ 0  1,  0 -1];
```

```
P = [0 +0.95, 0 -0.95];
```

```
vector_teta = geo_teta_z(zeros_in, polos_in, 240);
```

```
plot(vector_teta);
```

Veja Também:

plota\_tela\_Z, geo\_modulo\_z e eq\_diff

Autor: Diego Laucsen da Rosa

**Normalizateta = Melhora a visualização da resposta em fase.**

Sequencia de Chamada:

`Res = normalizateta(vector_teta);`

Parâmetros:

`vector_teta` = Vetor resposta da função `geo_teta_z`.  
*Res* = *Resposta da função*.

Descrição:

Melhora a visualização da resposta em fase do sistema para que os valores fiquem sempre dentro de  $+\pi$  ou  $-\pi$ .

Veja Também:

`geo_teta_z`

Autor: Diego Laucsen da Rosa

**plota\_tela\_Z – Plota Tela Z**Sequência de Chamada:

```
plota_tela_z(P,Z,vector_modulo,vector_teta)
```

Parâmetros:

P: Lista de Polos. Ex:  $P = [a \ b; \ c \ d]$ , onde  $a$  e  $c$  são a parte real e  $b$  e  $d$  a imaginária.

Z: Lista de Zeros. Ex. No mesmo modelo dos Polos.

vector\_modulo: Vetor de qualquer tamanho contendo o resultado da função que calcula o módulo. *Vide: geo\_modulo\_z*

vector\_teta: Vetor de qualquer tamanho contendo o resultado da função que calcula a defasagem do sistema. *Vide: geo\_teta\_z*

Descrição:

Plota em uma figura com quatro sub gráficos.

No canto superior esquerdo encontra-se o plano Z, o círculo unitário e todos os Polos e Zeros do sistema.

No canto superior direito encontra-se a magnitude do sinal.

No canto inferior esquerdo encontra-se a magnitude em Decibéis.

No canto inferior direito encontra-se a defasagem do sinal.

Exemplos:

```
fs = 1000; //1000 Hz
```

```
Z = [ 0 1, 0 -1];
```

```
P = [0 +0.95, 0 -0.95];
```

```
vector_modulo ???
```

```
vector_teta = geo_teta_z(zeros_in,polos_in,fs);
```

```
ta_tela_Z(P,Z,vector_modulo,vector_teta)
```

Veja Também:

geo\_teta\_z, geo\_modulo\_z e eq\_diff

Autor: Diego Laucsen da Rosa

**convlution = Convolução**

Sequencia de Chamada:

y = convlution(X,H);

Parâmetros:

X = Sinal de entrada

H = Sinal 2, ou o sistema pelo qual o sinal passa.

Y = Resultado da Convolução

Descrição:

Calcula a convolução entre X e H.

Autor: Diego Laucsen da Rosa

**filtro = cria\_filtro(tipo,banda,janela,N,fs) – Cria um Filtro FIR**

Sequencia de Chamada:

filtro = cria\_filtro(tipo,banda,janela,N,fs)

Parâmetros:

tipo = Tipo do Filtro.

- 1 – Passa Alta
- 2 – Passa Baixa
- 3 – Passa Banda
- 4 – Rejeita Banda

banda = É a frequência de corte (fc) ou trecho de frequências.

[0 fc] – Para o passa baixa: Frequência de corte é fc.

[fc fs] – Para passa alta: Frequência de corte é fc até a frequência de amostragem dividido por 2.

[fc1 fc2] – Para ao passa banda e rejeita banda: Deixa passar ou rejeita frequência entre fc1 e fc2, respectivamente.

janela = É o tipo da janela escolhida para o filtro.

- 1 = Retangular
- 2 = Triangular
- 3 = Blackman
- 4 = Hanning
- 5 = Hamming

N = Número de coeficientes do filtro.

fs = Frequência de amostragem.

Filtro = Variável de retorno possuindo os coeficientes para o filtro configurado.

Descrição:

Cria o vetor de coeficientes para o filtro configurado nos argumentos de entrada.

Para utilizar os resultados apenas realize uma convolução entre esses coeficientes e o sinal desejado.

Exemplos:

Ver exemplos 1,2,3 e 4.

Veja Também:

convlution, janelamento

Autor: Diego Laucsen da Rosa

**get\_att – Diferença de picos**

Sequencia de Chamada:

atenuacao = get\_att(sinal,tolerancia)

Parâmetros:

sinal = Sinal que se deseja analisar.

tolerancia = Tolerância. Quando n's serão cortado do começo e do fim do sinal. É importante considerar, pois um filtro com N coeficientes possivelmente leve N interações para estabilizar.

atenuacao = Valor da diferença

Descrição:

Calcula a diferença entre o pico mais alto e o pico mais baixo de um sinal, levando em consideração uma tolerância para corar trecho de estabilidade de filtro em sinais.

Vejá Também:

cria\_filtro e varredura

Autor: Diego Laucsen da Rosa

**janelamento - Distância Geométrica para o Plano Z**

Sequencia de Chamada:

janela = janelamento(numero\_daJanela,N)

Parâmetros:

numero\_daJanela – Número da janela escolhida

1 = Retangular

2 = Triangular

3 = Blackman

4 = Hanning

5 = Hamming

N – Número de coeficientes da janela

janela – Resultado, um vetor contendo a janela.

Descrição:

Calcula a janela de coeficientes para filtros FIR.

Ver cria\_filtro para janelas.

Veja Também:

cria\_filtro

Autor: Diego Laucsen da Rosa

**varredura – varredura de frequência**Sequencia de Chamada:

vetor = varredura(H,fs,resolucao)

Parâmetros:

H = Filtro configurado usando a função cria\_filtro.

fs = Frequência de amostragem

resolucao = Resolução do filtro. Por exemplo, com 1, ela cria uma frequência pra cada valor de fs. Para 2, cria 1 frequência de análise para cada 2 incrementos de fs. Quando maior a resolução, menor a precisão no gráfico, porém menor o processamento.

vetor = Vetor contendo a resposta em frequência para o filtro H.

Descrição:

Cria um vetor contendo a resposta em frequência de um filtro criado usando a função cria\_filtro.

Exemplos:

Ver Exemplos 1, 2, 3, 4

Veja Também:

cria\_filtro e get\_att

Autor: Diego Laucsen da Rosa



**HZ = BZT(Polos,Zeros,Fs)**

Sequencia de Chamada:

HZ = BZT(Polos,Zeros,Fs)

Parâmetros:

Polos = Polos do sistema no plano S.

Zeros = Zeros do sistema no plano S.

Fs = Frequência de amostragem.

HZ = Polinômio que representa a equação no plano Z.

Descrição:

Converte um conjunto de polos e zeros do plano S para o plano Z.

Exemplos:

Ver Exemplos.

Veja Também:

BZT\_ImagCalc e root2

Autor: Diego Laucsen da Rosa

## **HZ = BZT\_ImagCalc – Aplica a realização numérica da transformação Z Bilinear Z**

Sequencia de Chamada:

$HZ = BZT\_ImagCalc(ima,fs)$

Parâmetros:

ima = Número imaginário.

EX: ima = [0.5 0.1];

fs = Sempre 1. É o Td, geralmente encontrado nas expressões da Transformação.

HZ = Equação polinomial que representa a versão no plano Z dos pontos de entrada: Z – P, onde P é o ponto no plano Z equivalente a entrada. Para descobrir o ponto deve-se achar a raiz da resposta, para isso utilizar a função root ou root2. A root2 é melhor para a compreensão pois a mesma independe de valores complexos.

Descrição:

Converte um numero imaginário do plano S para o plano Z.

Exemplos:

Ver Exemplos

Veja Também:

BZT e root2

Autor: Diego Lauschen da Rosa

## **Raízes Polinomiais de Segunda Ordem 2**

**HZ = root2(eq)**

Seqüência de Chamada:

HZ = root2(eq)

Parâmetros:

eq = Equação de segundo grau.

HZ = Raízes de um polinômio de segundo grau.

Descrição:

Calcula as raízes de um polinômio de segundo grau. O diferencial da root, do Scilab, é que em caso de respostas em que as raízes são números imaginários, a parte real virá separada da imaginária, em uma matriz contendo em cada linha uma raiz, e na primeira coluna a parte real e na segunda coluna a parte imaginária.

Exemplos:

Ver Exemplos.

Autor: Diego Laucsen da Rosa

## REFERÊNCIAS BIBLIOGRÁFICAS

AGOSTINI, M.; SILVEIRA, F.; ZURN, H. Normas para apresentações de dissertações e teses. Universidade Federal de Santa Catarina. Florianópolis. Centro Tecnológico, 1999.

ANDRIGHETTO, E. **Sistema de Processamento de Sinais Biomédicos: Rede Wireless Zigbee com Aplicação do Padrão IEEE 802.15.4.** Universidade Federal de Santa Catarina. Florianópolis. Instituto de Engenharia Biomédica, 2008.

BRIESE, R. G. **Sistema de Processamento de Sinais Biomédicos: Módulo de Estudo de Filtros Analógicos.** Universidade Federal de Santa Catarina. Florianópolis. Instituto de Engenharia Biomédica, 2009.

CADZOW, J. A. H.R. **Martens, Discrete-Time and Computer Control Systems.** Prentice-Hall, inc. Englewood Cliffs. New Jersey, 1970. p. 473.

DIEDERMEYER, E.; Silva F. L.; **Electroencephalography: Basic Principles, Clinical Applications and Related Fields.** Quinta Edição, Ilustrada. Lippincott Williams & Wilkins, 2004. pg. 1309.

IFEACHOR, E. C.; JERVIS, B. W. **Digital Signal Processing.** Addison-Wesley Publishing Company Ltd, 1993. p. 760.

**JAVA; Disponível em <[http://java.com/pt\\_BR/about/](http://java.com/pt_BR/about/)>. Acesso em 26 Nov. 2008.**

**JUDE – Design & Modeling Tool; Disponível em <<http://jude.change-vision.com/>>. Acesso em 26 Nov. 2008.**

KOOTSEY, J. M.; MCAULEY, G. e LIU, H. **Presenting Systems Concepts in Physiology and Pharmacology with Simulation Applets in Java.** 23rd Annual EMBS International Conference. October 25-28. Istanbul. Turkey, 2001.

MEDEIROS, E. S. **Desenvolvendo Software com UML 2.0, Definitivo.** São Paulo, Pearson Makron Books, 2004. p. 264.

OPPENHEIM A. V.; SCHAFER R. W. **Discrete-Time Signal Processing.** Prentice Hall. Upper Saddle River. New Jersey, 1999.

POSSA, P. R. D. **Sistema de Processamento de Sinais Biomédicos: Módulo Didático de Amplificador de Potenciais Bioelétricos.** Universidade Federal de Santa Catarina. Florianópolis. Instituto de Engenharia Biomédica, 2008.

POSSA, P. R.; ANDRIGHETTO, E. et. al. **A inserção da Engenharia Biomédica nos cursos de graduação de Engenharia Elétrica no Brasil**. Revista Brasileira de Engenharia Biomédica, v. 24, n. 2, p. 99-108, agosto, 2008.

SAHARIL, F.; HASBI, A. M. E HUSSIN H. **Educational Java Applets for MOS Capacitor**. International Conference on Software Engineering. Kaula Lumpur. Malaysia, 2004.

\_\_\_\_\_. **Scilab, The Open Source Platform for Numerical Computation**. Disponível em <<http://www.scilab.org>>. Acesso em 26 Nov. 2008.

SHENOI, B. A. **Introduction to Digital Signal and Processing and Filter Design**. John Wiley & Sons, Inc. Hoboken. New Jersey, 2006. p. 423.

SMALL, R. V. e ARNONE M. 1999. **The Motivation Mining Company; Website Motivational Analysis Checklist Senior (WebMac 4.0)**. Disponível em <<http://www.marilynarnone.com/WebMACSr.4.0.pdf>>. Acesso em: 27 Nov. 2008.